GET SERIOUS Pi 4 COOLING | MAKE A KODI Pi MEDIA CENTRE

LINUX FORMAT

The #1 open source mag

Create perfect 3D printer models!

How to save your dying Smart Home devices

MULTI-BOOT MINT 21!

Install the new Linux Mint on your PC and boot it with anything – Ubuntu, Fedora... even Windows!

PLUS: HOW TO

- Create better cutouts in GIMP
- Fire up modern open Solaris
- Use your phone as a webcam

PACKET FILTERS
Using the Linux eBPF to monitor networks

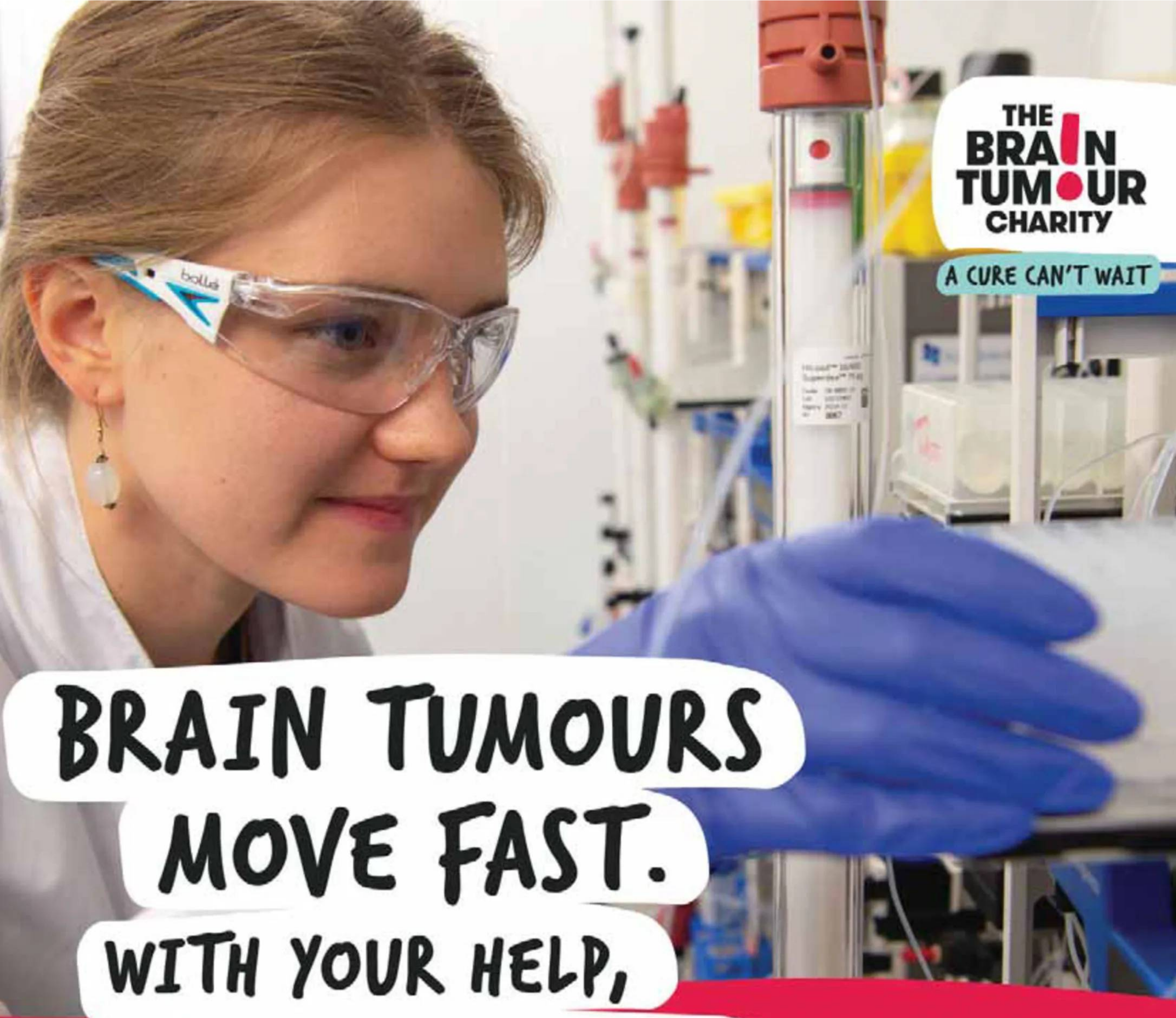
ALMA LINUX 9.0
We explore the distro replacing CentOS

CODING PAC-MAN
Learn Python to create the gaming classic!

LXF October 2022

FUTURE

3kv8CGgdIRsHphQFjEgfZw==


THE BRAIN TUMOUR CHARITY
A CURE CAN'T WAIT
BRAIN TUMOURS MOVE FAST.
WITH YOUR HELP, WE CAN TOO!

We're working to create a future where brain tumours are curable. We urgently need your help to accelerate research.

Text DEFEAT5 to 70507 to donate £5, please help us to find a cure.

thebraintumourcharity.org

© The Brain Tumour Charity 2020. Registered Charity in England and Wales (1150054) and Scotland (SC045081)

Registered with FUNDRAISING REGULATOR

3Jw8GGqdiRsHphQFjEgfZw==

MEET THE TEAM

This issue we're seeing how many operating systems Jonni can run on one PC. We asked our writers what's the most OSes they've had on a system, and why.

Jonni Bidwell
Well, the poor PC that was forced to accommodate seven distros for the cover feature could probably fit another one on. Or we could try installing Windows on it. Methinks then there would be zero operating systems – working ones, anyway.

Nick Peers
I've had numerous dual-boot systems in my time, including one MacBook configured to dual-boot macOS and ElementaryOS. But with ever-increasing processor cores and RAM, virtual is the way to go – I'm frequently running multiple Linux OSes on my 32GB Ryzen 2700 PC...

David Rutland
I don't dual-boot, and I never have. One time I had Windows running inside a VM on my Ubuntu laptop. Within the Windows VM, WinUAE provided me with a virtual Amiga on which I ran *AmigaVGB* – a Gameboy emulator. Why? I wanted to see if it would work.

Michael Reed
Setting up a dedicated machine as a Linux server is a valid approach, but don't forget to consider consolidation. That is, rather than having separate computers, you could add extra resources such as RAM to your main computer and add the servers using virtualisation and containerisation.

Mayank Sharma
I've never run more than a couple of physical installations, with one of them sat inside a perpetual test partition. The real churn for me takes place inside VMs, and besides three permanent, updated ones, I haven't had more than eight running concurrently, be it for reviews or an article.

Flexible friend

Things used to be simple: you'd have a master boot record that pointed to a bootloader and away your operating system went. But, of course, people wanted more features and something called security, so LILO appeared and then Grub, and Windows had BCD and Secure Boot was added along with TPMs and now there's Pluton. Yet even with all the new layers and security there's no reason why you can't multi-boot as many operating systems as you like, and we're here to show you how!

We're also using the release of Linux Mint 21 as an excuse on how to get it installed alongside the popular rectangular aperture OS from Microsoft. In a final attempt to see off his ancient all-in-one PC Jonni's installed seven operating systems booting alongside each other, and annoyingly it's all working smoothly. Find out the grim story of how that happened further inside...

If you've had enough Grub there's plenty more topics to dig into this issue, from fostering your creative side to producing better-quality cutouts in *GIMP*, perfecting your 3D prints and the best writing tools, while more geeky subjects include exploring the OpenIndiana open source OS, hiding messages inside images, writing your own *Pac-Man* game and taking your first steps with the eBPF kernel system, so enjoy!

Neil

Neil Mohr Editor
neil.mohr@futurenet.com

Subscribe & save!
On digital and print – see p16

Contents

SUBSCRIBE NOW!
Page 16

REVIEWS

AMD Ryzen 5500 19
Too little, too late is what **Alan Dexter** hears a lot, but at least this time it's nothing to do with his cooking and everything to do with the performance of AMD's latest CPU.

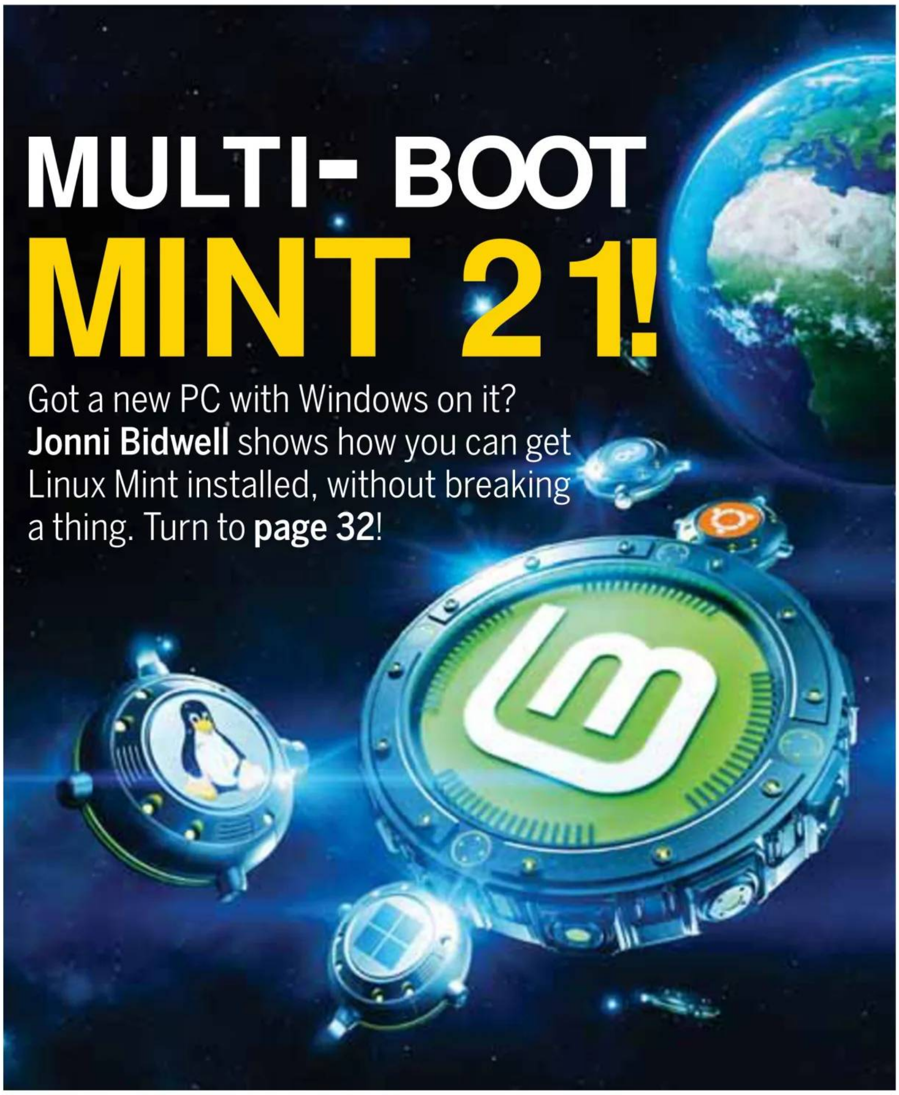
DragonflyBSD 6.2.2 20
David Rutland takes a long and confusing journey – during which he might annoy BSD users – to the land before time, where things look familiar… but aren't.

KDE Neon 2022.08 21
David Rutland finds that KDE is a delight to behold and regardless of whether it's a 'real' distro, it's a charming and relaxing place to do his Linuxing.

NuTyX 22.07 22
David Rutland finds this distro with its unique package management system a little less nutty than he had expected, but it's certainly a bit crunchy…

Emmabuntüs 4-1 23
David Rutland finds that the latest stable distro release is a little lacking in the "zig-a-zig-ah" department. Maybe it was lost in translation…

Psychonauts 2 24
When Management said they're implementing a mindfulness policy, **Matthew Castle** didn't realise it meant entering his very psyche!



MULTI-BOOT MINT 21!

Got a new PC with Windows on it? **Jonni Bidwell** shows how you can get Linux Mint installed, without breaking a thing. Turn to **page 32**!

ROUNDUP

Creative writing tools 26
Struggling to compose a story? **Nick Peers** reviews five tools that all provide more than a simple writing environment to help bring your novels to life.

IN DEPTH

Saving the smart home 50
The smart home is dying, but **Matt Holder** is here to save the day by showing us how open source can keep the smart lights on and resurrect your dead devices…

Pi USER

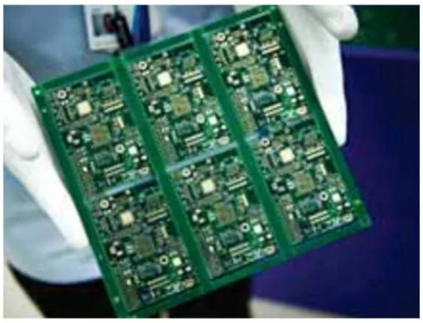
Raspberry Pi news 41
Les Pounder pays tribute to Olly Clark, a key member of the Pi community, and why the RS Group has lost its Pi licence.

Low-Profile ICE Tower 42
Could **Les Pounder's** new cooler make even our expert chill out over summer?

Pimoroni Trilobot 43
Les Pounder makes a new friend with a powerful and affordable Pi-powered robot.

Stream live data to web pages 44
Les Pounder takes the latest and smallest Raspberry Pi with Wi-Fi and turns it into a £6 web server with live sensor data.

Make a Pi Media Centre 46
We test drive the latest Kodi build so **Michael Reed** can get some serious couch-based entertainment.



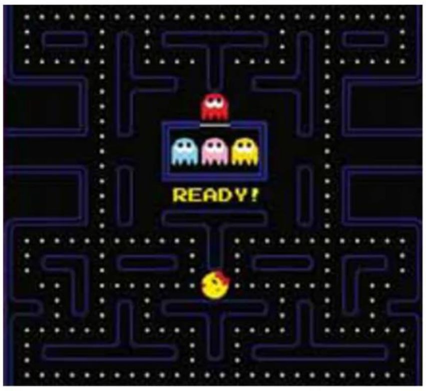
KODI
QUESTOR

CODING ACADEMY

Recreate Pac-Man in Python 88
Being chased down endless corridors by spectral apparitions is all in a day's work for **Andrew Smith** and his Python skills...

Add dialogs to terminal scripts 92
Quickly create dialogs in shell scripts with YAD and a bit of **Karsten Günther**!

Get to grips with the complex eBPF 94
Extending things is what **Mihalis Tsoukalos** loves to do, so his heart leapt at the chance to explain the Berkley Packet Filter..



REGULARS AT A GLANCE

News 6
Debian switches to DuckDuckGo, but should it? Turns out tractors can play *Doom*! There's drama in PINE64 land! Ubuntu is on RISC-V. A new *LibreOffice* and FlatHub updates.

Kernel watch 10

Answers 11
How to streamline a bloated Debian installation, printing PostScript files, recommending a video player for Linux, and when are cloned files not exact copies?

Mailserver 14
Does anyone out there know about DMX lighting software? How about writing for Fedora? And did you know we make the odd mistake, too? Shurely shome mishtake?!

Subscriptions 16
Get your monthly Linux dose and save cash!

Back issues 62
Get hold of previous *Linux Format* editions.

Overseas subscriptions 63
Get *Linux Format* shipped around the globe.

HotPicks 81
Mayank Sharma hasn't been firing missiles into neighbouring country by accident – he's too busy firing off downloads and compilation tasks of top FOSS such as *Syncthing, Imagine, HomeBank, Jitsi, Beets, vifm, Rclone, Unvanquished, Bitfighter, AES Crypt* and *SMPlayer*.

Next month 98

TUTORIALS

TERMINAL: Kmon 54
There's a lot to be said about finding inner peace through meditation. Yet **Shashank Sharma** is confident he'll find nirvana with this classy CLI tool.

DROIDCAM: Smart-phone webcam 56
Nick Peers reveals how to convert your mobile phones and tablets into live cameras you can use for chatting or monitoring with Linux.

OPENINDIANA: Modern Solaris 58
Linux veteran **Michael Reed** wonders if he can pass for a native in the weird and wonderful land of Solaris-derived OpenIndiana or will he be caught out...

CURA: Perfect 3D prints 64
Simply not getting the quality of prints you expected from your 3D printer? **Ali Jennings** shows you the essential steps the manuals often miss..

GIMP: Creating better cutouts 68
There are many situations in which parts of an image need to be "isolated" from the background. Use *GIMP* and **Karsten Günther** for this!

STEGANOGRAPHY: Hiding messages 72
Steganography might not be as well-known as cryptography but, as **Mike Bedford** reveals, it's a powerful tool in preserving your security.

TOP OF THE FOSS!

Inside AlmaLinux 76
Andrew Lukoshko, benny Vasquez and Jack Aboutboul update **Jonni Bidwell** on the latest AlmaLinux developments.

Newsdesk

THIS ISSUE: Debian picks DuckDuckGo » GPL non-compliance » Setback for PINE64 » Greater Canonical support for RISC-V

SEARCH

Debian embraces DuckDuckGo

The distro's switch from Google as its default search engine causes controversy over privacy concerns.

DuckDuckGo.

DuckDuckGo was once the go-to search engine for privacy-conscious users, but a deal with Microsoft made it fall out of favour with some.

Usually when a distro announces that it's ditching Google for another default search engine, privacy campaigners – and anyone concerned about how their online activity is tracked and shared – would be pleased. However, Debian has managed to provoke more concern after it announced it was switching from Google because of the alternative search engine it's chosen: DuckDuckGo.

While DuckDuckGo was once touted as a privacy-focused search engine, it caused outrage when it emerged the team had an agreement with Microsoft that allows search content data to be shared between the companies, and enables Microsoft trackers to run on third-party sites.

Security researcher Zach Edwards found that while trackers from Google and Facebook are indeed blocked by DuckDuckGo, Microsoft trackers are permitted, sending data to the company's services such as LinkedIn and Bing. You can read Edwards' original tweet thread regarding his findings at **https://bit.ly/lxf294ddg**. Despite a vocal defence of the agreement by DuckDuckGo CEO and founder Gabriel Weinberg, who pointed out that this only affected the *DuckDuckGo* browser and not the search engine itself, the damage was done in the eyes of many people.

So, when the Debian team announced (**https://bit.ly/lxf294debian**) that the default search engine in its *Chromium* browser would be changed to DuckDuckGo for "privacy reasons" many people were upset at the choice. The idea to replace Google with DuckDuckGo has actually been bobbing around for a while now, with references to it spotted back in April 2020 in a bug report. While the team clearly took their time making the move, and Weinberg again did the rounds (including posting on *Hacker News* under the user name Yegg at **https://bit.ly/lxf294yegg**), many people are still wary, despite assurances about their privacy, and the fact that *Firefox* remains the default browser in Debian, not *Chromium* (and even then, it's easy to change the default search engine to something else).

The fact that people are still concerned about this move goes to show just how much DuckDuckGo's agreement with Microsoft has damaged the search engine's reputation. There are also some unintended side-effects of this switch, with one user reporting on *Hacker News* (**https://bit.ly/lxf294hackernews**) that when the switch happened, they were alerted that their *Chromium* browser was now being managed by their "organisation", causing them to worry that they had been compromised – with other users reporting the same. For a relatively minor change, this has managed to annoy a lot of people.

> **WHY SOME PEOPLE ARE ANGRY…**
>
> "While trackers from Google and Facebook are indeed blocked by DuckDuckGo, Microsoft trackers are permitted."

HACKING

Hacked tractor plays Doom…

…but also runs outdated and unpatched hardware.

We always love seeing iconic PC game *Doom* running on surprising hardware, and the talented hacker known as Sick Codes (**https://bit.ly/lxf294sickcodes**) recently did just that at the recent Def Con event (**https://defcon.org**). However, it wasn't just for fun, because there was an important message with the project.

It was also a lot of fun, however. Sick Codes hacked the touchscreen interface of a John Deere tractor to play a special farm-themed mod of *Doom* (made by **@Skelegant** – see more of their work at **https://bit.ly/lxf29skelegant**), which includes fighting pigs and harvesting corn to replenish your health. Very appropriate.

However, it also revealed that the US-based tractor manufacturer was playing fast and loose with software under the GPL compliance. The idea that John Deere is making money from open source software is particularly galling, because the company has been at the centre of a right-to-repair controversy by making it all but impossible for owners of its tractors to repair their products due to proprietary software. By disclosing the hack, Sick Codes has helped farmers gain more control over their expensive purchases.

Perhaps most worrying is that the ecosystem used for John Deere tractors is based on outdated Linux and Windows CE hardware, much of which is unpatched.

This whole thing has meant that John Deere's standing with the open source community, as well as anyone who believes that consumers have a right to repair the products they purchase, is now at its lowest. It also shows how important these white hat hacking events can be in highlighting secrets some companies may rather were kept hidden.

No Name
100%
0%

Hacker Sick Codes programmed a tractor to play a farming-themed mod of Doom. CREDIT: @sickcodes, Twitter

OPEN HARDWARE

PINE64 drama as moderator quits

Publicly calls out PinePhone manufacturer's failings.

We've often talked about PINE64's projects, including the PinePhone and PineBook. The company caught our eye by producing devices with an admirable commitment to both open source software and hardware, and with the PinePhone brought us a Linux-powered smartphone when other projects, such as Ubuntu Touch, failed.

However, there's been a bit of ugly drama, with Martijn Braam, a moderator for PINE64, publicly quitting and writing a lengthy blog post (**https://bit.ly/lxf294pine64quit**) about why. It's not easy working for the PINE64 team, with Braam citing the decision by PINE64 to switch from supporting a range of distros, to focusing on Manjaro. The last straw for him was a change to how the PinePhone Pro boots, which forces Manjaro to boot from eMMC storage, rather than enabling users to choose and boot a distro from a microSD card.

While Braam states that "after months of discussions between the community, Manjaro, and PINE64 leadership, we finally were able to convince them to ship the PinePhone Pro with an SPI flash chip with Tow-Boot installed on it," allowing other distros to boot without having to rely on U-Boot firmware, he was dismayed to find that the Pinebook Pro would ship with a similar boot constraining boot configurations, and no Tow-Boot included.

Braam claims PINE64 is neglecting its users in a bid to curry favour with Manjaro. This has caused PINE64 to issue a defence (**https://bit.ly/lxf294pine64response**). Hopefully this drama doesn't distract from the good things this project has achieved.

OPINION

STOP THE SPRAWL

Donnie Berkholz
is senior vice-president, product at Percona

" According to StackOverflow's survey for this year, cloud platforms take up the top three slots in the 'Most Wanted' category. Building cloud native applications means thinking about how to use distinct microservices for each function and connect them up efficiently. Using the cloud, you can scale those services up as you need them, based on containers. Google's Kubernetes is now the default for this, enabling you to automate all those tasks.

This approach is great when it helps you build new services quickly. However, when you have individuals all using their own preferred services, it's easy to suffer from sprawl. This soon becomes unwieldy over time, as well as becoming more expensive to run.

To get the most out of the cloud it's important to collaborate on what that framework should be over time. By standardising on specific open source databases and other application components, you can stop sprawl from taking place. You can also define when new defaults are needed, or where edge cases demand another service.

We need to consolidate where we can, ensure our sprawl is kept in check, and use management tools to cover all those platforms in one place. "

OPINION

MANAGING EXPECTATIONS

Keith Edmunds is MD of Tiger Computing Ltd, which provides support for businesses using Linux.

“We took on some new (to us) Linux servers onto support recently. Both the client and we knew that there was a lot of work needed to make them secure and reliable.

It can be daunting when the work required looks like a mountain, and you’re standing at base camp surrounded by piles of equipment. It’s also very easy to fall into the trap of thinking, “While we’re doing all that work, we might as well [insert additional work tasks here]”.

Of course it’s a mistake to think like this. A much better approach would to be identify a few key critical issues and resolve them. Yes, that still leaves other issues to be resolved, but already the servers are in better shape. No long downtime. No big expense for the customer.

So that’s what we did. The client is happy that things are progressing in the right direction and is already benefiting from some quick wins. Interestingly, with those wins in place, the next steps have subtly changed.

The goal need not be perfection. Typically a goal of “better than it is now” is enough. Repeating that process often enough gets us to the end point, and often more efficiently, too.”

HARDWARE

Canonical embraces RISC-V

Ubuntu now supported on StarFive’s VisionFive RISC-V boards.

VisionFive

Credit: StarFive Technology

The VisionFive single board computer runs on RISC-V architecture, and now supports Ubuntu.

Canonical announced in 2021 RISC-V compatibility for Ubuntu, enabling its popular Linux distro to run on the open-source processor architecture. It’s now increased support by announcing an Ubuntu release for the StarFive’s VisionFive board. This single-board computer runs on the JH7100 SoC, which is based on a 64-bit RISC-V dual-core processor that runs at 1GHz, making it one of the more powerful RISC-V boards out there.

Along with 2MB L2 cache and 8GB of RAM, it should be able to run Ubuntu well, and as the Canonical’s blog post about the new support states (**https://bit.ly/lxf294ubunturisc**), “The availability of Ubuntu running on the VisionFive board comes as the result of the joint work between Canonical and StarFive’s engineering teams”, and an image of Ubuntu 22.04.1 for the VisionFive board can be downloaded from **https://bit.ly/lxf294ubuntudownload**.

StarFive, the Chinese company that specialises in creating RISC-V projects, will be showcasing Ubuntu support on the VisionFive at this year’s RISC-V China summit (**https://riscv-summit-china.com**). Single board computers like the VisionFive and Raspberry Pi are great for introducing new generations to Linux, so this additional support from Canonical is welcome.

APPLICATIONS

Flathub undergoes key changes

The project will make donating to teams easier.

Flathub (**https://flathub.org**), the base for Flatpak, is one of the most popular methods of distributing programs for Linux, no matter what distro you’re running, and it aims to make donating to developers quick and easy. This enables users to show their appreciation while helping to reward developers – and also encourage them to keep their programs updated and maintained.

Codethink, a team of open source experts working on the project (**www.codethink.co.uk**) recently revealed what the project has achieved since it joined in February 2022, including “a login flow for developers via GitHub, GitLab, and Google,” and a *Stripe*-powered donation system, which can be seen on the Flathub beta website at **https://beta.flathub.org**.

The blog post (**https://bit.ly/lxf294flathub**) also talks about what’s in store for the project. The donation infrastructure is set to be boosted, with program pages having their own links for donations, and developers can verify their work via GitHub or GitLab to ensure they get paid.

OFFICE SUITE

LibreOffice 7.4 released

Now with greater file support and documentation for scripts.

LibreOffice 7.4 Community is now available to download from **https://bit.ly/lxf294libreoffice**, with some useful new features. The latest version now supports WebP (*but Effy still doesn’t–Ed*) images and EMZ/WMZ files, adds a search field for the Extension Manager and improves the documentation for the ScriptForge scripting library.

Elsewhere, *Writer*, the suite’s word processing program, benefits from improved change tracking and new typographic settings for hyphenation, while *Calc*, the spreadsheet application, supports up to 16,384 columns and adds extra functions to the AutoSum widget. Meanwhile, *Impress*, the slideshow application, receives support for document themes.

LibreOffice 7.4
The Document Foundation

LibreOffice 7.4 brings new features and tweaks.

Distro watch

What's behind the free software sofa?

MX LINUX 21.2

This second refresh of MX Linux 21 has been released, and while it's still based on Debian 11 and comes with the Linux kernel 5.18, it also features several important bug fixes and updates. The base is Debian 11.4 "Bullseye" and with this release *mx-tweak* now makes it easy to disable Bluetooth, and features user interface improvements, and bugs have been fixed in *mx-installer*. If you're already running MX Linux 21 then you don't need to reinstall – just check the update channel. To find out more, see the release announcement at **https://bit.ly/lxf294mxlinux**.

MX Linux is a popular distro that uses the Xfce desktop environment.

MABOX LINUX 22.08

Mabox Linux is a rolling-release distro based on Manjaro and with a customised *Openbox* window manager, and the latest refresh is now available to download. It comes with the LTS Linux kernel 5.15 (as well as an option to use the older Linux kernel 5.4 LTS), and features a new tool called *Colorizer*. This makes it possible for you to easily adjust the colour of elements that make up the desktop via dynamically generated menus. Find out more at **https://bit.ly/lxf294mabox**, along with screenshots showing off *Colorizer* in action.

Mabox Linux features a nifty new tool that can make your desktop sparkle.

NEPTUNE 7.5

This new release of the Debian-based distro aims to improve hardware support and improve performance across the board. The kernel has been updated to 5.18, and performance has been tweaked to boost write speeds on USB thumb drives, and there's a new backup program, too – *Timeshift* – which enables you to make a copy of your entire system for safe keeping. These snapshots can help you recover your system, even if it appears to be completely broken. Visit **https://bit.ly/lxf294neptune** for more information.

Neptune is a distro with a USB installer and easy-to-use multimedia playback.

Q4OS 4.10

Another Debian-based distro has been updated, again now based on 11.4 "Bullseye". The Trinity desktop environment has been included, with the latest version preinstalled after what the team describes as a "careful review and testing" period. You can find out more about Trinity at **https://bit.ly/lxf294trinity**. The latest version of Q4OS also comes with an updated stable kernel, along with important security updates. To find out more about this release, and to download it, head over to the Q4OS blog at **https://bit.ly/lxf294q4os**.

Q4OS's user interface, Trinity, should be familiar to Windows users.

OPINION

AI GETS IN YOUR FACE

Marcus Edel
is a senior software engineer at Collabora

Over the past few years, different video codecs have been developed to meet the needs of various applications. The quality of the reconstructed video using these codecs is excellent at medium-to-low bitrates, but it degrades when operating at very low bitrates. While these codecs leverage expert knowledge of human perception and carefully engineered signal processing pipelines, there has been a massive interest in replacing these handcrafted methods with machine-learning approaches that learn to encode video data.

Using open source software, Collabora has developed an efficient compression pipeline that enables a face video broadcasting system that achieves the same visual quality as the H.264 standard while only using one-tenth of the bandwidth. The face video compression algorithms rely on a source frame of the face, a pipeline to extract the important features from a face image, and a generator to reconstruct the face using the extracted and compressed features on the receiving side.

Following an open source approach, the code is freely available and it all runs on a completely open-source stack.

OPINION

MY SPACE

Jon Masters is a kernel hacker who's been involved with Linux for more than 22 years, and works on energy-efficient Arm servers.

“I moved house earlier this year. Like many others, the pandemic changed life for me and my wife, and in some ways for the better. I had traditionally thought of myself as a “city cat”, and for the better part of several decades I lived right in the middle of Cambridge, Massachusetts. But in the past few years, I've realised that I don't need to be in the middle of town any more, and that further out means the usual trade-offs enabling more space, including for geekery.

As a result, we bought a house with enough room for a basement lab expansion. For the first time I have a dedicated space that's not also my home office.

But now I have the new space, and with it a need to update my lab automation. For many years, I've run an installation of *Beaker*, the Open Source lab automation software originally created by Red Hat for its own labs. What I like about *Beaker* is that I can attach automatic power cyclers, serial consoles (via a console server), and more. The net result is that I can go to a single “inventory” webpage and single-click install a machine with a Linux distro.

If you haven't played with such automation, I can recommend it, either for a somewhat overdone home lab, or for a corporate dev environment.”

Kernel Watch

Jon Masters summarises the latest happenings in the Linux kernel, so that you don't have to.

Linus Torvalds announced Linux 6.0-rc4 (Release Candidate 4) following the earlier closure of the “merge window” for what will become the Linux 6.0 kernel in a few more weeks. Last issue we speculated that Linus was likely to go with the “6.0” naming for this cycle, and that turned out to be right. As Linus himself said, however, “Despite the major number change, there's nothing fundamentally different about this release – I've long eschewed the notion that major numbers are meaningful, and the only reason for a ‘hierarchical' numbering system is to make the numbers easier to remember and distinguish.”

On a lighter note, it's well known that Linus likes to keep kernel minor numbers to roughly what can be counted using all of his “fingers and toes”, but he also notes that in Chinese culture “5.20” has a “wholesome” meaning [it's the equivalent of Valentine's Day since 520 sounds similar to “I love you” in Chinese]. Linus said: “If you want to call this ‘Linux 5.20', go right ahead. Because the kernel version numbers really are entirely made up.”

Features arriving in Linux 6.0 include the “runtime verification subsystem”, which will enable the kernel to self-check whether it's transitioning between two permitted states, potentially allowing for automatic detection of bugs. Perhaps what stands out just as much is what isn't happening: Linux 6.0 won't (yet) include Rust support. We'll have a complete summary of 6.0 next issue.

Printk gets a dressing down

Linus was angry with the “printk” maintainers in this cycle. The vulnerable “printk” functions are used within the kernel for logging messages to the console, as well as to the system log (for example, **/var/log/syslog** or – on more contemporary systems – journald's log viewable via *journalctl*). For example, `printk("Hello, World!\n");` in kernel code will cause `Hello, World!` to be logged by the kernel. But due to the need to be able to always log such messages, even in critical parts of the kernel, printk is special.

This can lead to problems, such as introducing unwanted latency when running a so-called “Real Time” Linux kernel. As a result, the printk maintainers recently decided simply to “Completely disable printing on consoles with CONFIG_RT”. This made Linus hit the roof, with “We don't suddenly change behaviour just because some random developer has decided ‘this is the RightThing™ to do'”. He rejected the patches for Linux 6.0, and then proceeded with an extremely lengthy rant, ending with, “I'm concerned that I will not be pulling printk patches in the future either, because of where these pull requests seem to be trending.” LXF

» ONGOING DEVELOPMENT

Jonathan Corbet (the editor of *Linux Weekly News*) announced the 2022 election for membership on the Linux Foundation Technical Advisory Board (TAB) will be held electronically during the 2022 Kernel Summit and Linux Plumbers Conference. The TAB serves as a backstop to handle thorny issues that the Linux Foundation needs advice in addressing, as well as enforcing the kernel Code of Conduct.

Ashok Raj (from Intel) posted version 3 of a patch series titled “Making microcode late-load robust”. Microcode is used on some (especially x86) processors to update CPU behaviour or to address issues (such as the side-channel vulnerabilities: Spectre, Meltdown and friends). But back in 2019 it was noted that Linux didn't robustly handle “late loading” (load of microcode after the system has booted) properly, and this was by default not enabled as a result. Ashok is attempting to remediate this and re-enable it. The upshot will be that users have more options for runtime ucode updates.

There have been a great many patches posted for the LoongArch, a Chinese architecture following in a similar vain to RISC-V as an open alternative to traditional options. The number of patches was notably higher this month, which could be an indication of the level of investment currently taking place. Among the patches was initial support for attaching PCI devices in Linux 6.0.

Answers

Got a burning question about open source or the kernel? Whatever your level, email it to **lxf.answers@futurenet.com**

Neil Bothwick loves printer problems, send more in, he'll love it!

Q Bloated Debian

This Debian 11 is my first attempt to explore Debian and I truly loved it for its stability and user-friendliness. I'm dual-booting Debian with Windows 10. I was wrong to set the Debian partition of 40GB only (2.6GB/41.8GB available). A short while later it seems the hard disk is full and many programs won't install. I've removed many unnecessary applications, yet I still can't install new ones.

What must I do to regain new space or to enlarge the partition?

Joe Wyatt

A In our view, 40GB is more than enough for even the most generous of installations, even with multiple desktop environments, so you must be losing space elsewhere. If you have set up a separate partition for **/home**, your home directory is the first place to look, especially if you've been downloading large files such as videos. If **/home** is not part of **/**, it's possible that something is generating a lot of error messages and filling up **/var/log** with them, although it would have to be a lot to use that much space.

We'd normally suggest installing a disk-space viewer, such as the terminal-based *ncdu* or one of the graphical desktop equivalents, but you can't install anything until you've at least partially resolved this problem. Fortunately, you can get a good idea of disk space usage with the standard tools you already have. `$ df -h` will show the usage of each filesystem, both in gigabytes and percentage. If your root partition isn't showing 100 per cent usage here, it's possible you've run out of inodes, which you can check with:

```
$ df -i
```

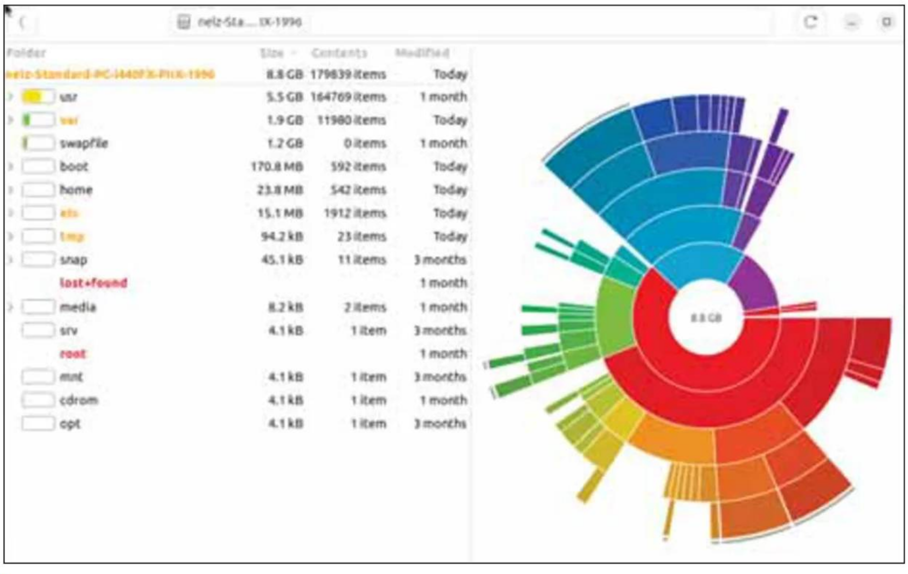
If this shows 100 per cent, you're looking for a large number of smaller files, possibly in **/tmp**. To see which directories are using the most space, run:

```
$ du -hx --max-depth=1 / | sort -h
```

The `du` command shows the disk usage of a directory and all the directories beneath it. The `-h` argument produces human-friendly figures while `-x` restricts the search to the current filesystem, ignoring mount points. To keep the output relevant, `--max-depth=1` shows a summary for each directory in the target. If your problem is a lack of inodes, as reported by `df -i`, replace `-h` with `-i` to shows this information. The output from `du` is then passed to the `sort` command, where `-h` tells `sort` to treat the lines as human-friendly numbers. The biggest consumers of space will appear at the bottom of the list. You can then take appropriate action.

If the total usage reported by `du` doesn't account for a full filesystem, it's possible you have files taking up space but not visible because another filesystem has been mounted over their location. You can work around this with a bind mount, but the easiest way is usually to boot from a live distro, mount your root partition and then run the disk usage tools from there.

You can resize partitions using the *GParted* partition manager, available on most live distros (resizing a running filesystem is a bad idea). However, you need to find out what is filling your existing filesystem first, or the problem will reoccur.



Folder	Size	Contents	Modified
neiz-Standard-PC-I440FX-PIIX-1996	8.8 GB	179839 items	Today
usr	5.5 GB	164769 items	1 month
var	1.9 GB	11980 items	Today
swapfile	1.2 GB	0 items	1 month
boot	170.8 MB	592 items	Today
home	23.8 MB	542 items	Today
etc	15.1 MB	1912 items	Today
tmp	94.2 kB	23 items	Today
snap	45.1 kB	11 items	3 months
lost+found			1 month
media	8.2 kB	2 items	1 month
srv	4.1 kB	1 item	3 months
root			1 month
mnt	4.1 kB	1 item	3 months
cdrom	4.1 kB	1 item	1 month
opt	4.1 kB	1 item	3 months

Graphical disk usage tools are an easy way to find out where your disk space has gone, as long as you have the disk space to install one.

Q PostScript, what PostScript?

I have Mint 20.3 and Peppermint 10 on two different machines. I'm trying to print to my HP P2015 printer or my HP 4550 Color LaserJet printer from *Scribus* from either machine. *Scribus* happily uses both printers with Windows 10. Linux *Scribus* insists in trying to print to these printer via PDF or PostScript. These printers aren't PostScript printers. As a result the output doesn't even register in the print queues, they just go to never-neverland.

Richard E Crislip

A *Scribus* is a desktop publishing program, and publishing generally uses PostScript output. This doesn't mean that you need a PostScript printer because the drivers generally take care of this. You haven't said which drivers you're using. I would expect it to be the HPLIP package because these are both HP printers, but you should be able to get more information about what's happening from the CUPS error log. In a terminal, run:

```
$ tail -f /var/log/cups/error.log
```

You may need to prefix tail with `sudo` if you obtain a permissions error when running the command. Then try to print a document and watch the terminal output. This should give a clue as to what's going on. We've successfully printed from *Scribus* to our HP Color LaserJet with no issues so it's likely to be either the wrong driver or unsuitable configuration. Press the options button next to the printer selection in *Scribus* to check that the settings here are suitable. You can also try different PostScript levels – level 3 has always worked for us.

There is a workround, albeit cumbersome and kludgy, that we've used in the past with uncooperative software. Use the Print to File option to create a PDF file, then print that from your PDF viewer.

Q RTSP Top Stream Player

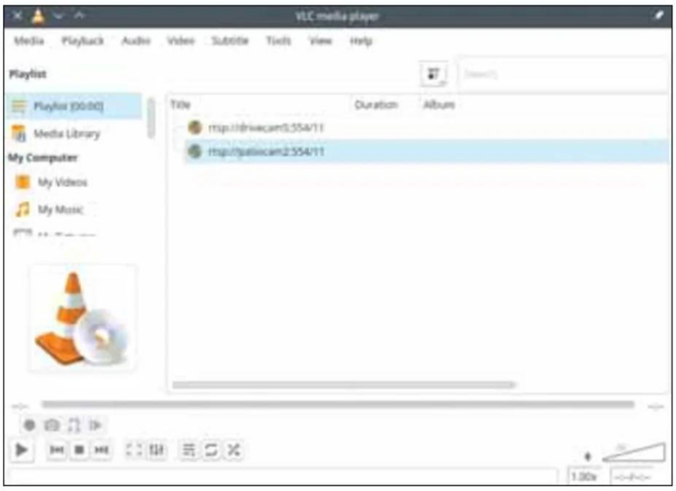
I've been using *RTSP Player* and *MX Player* for playing streaming video on my Android device. They're both very good players for RTSP. Are either *RTSP Player* or *MX Player* available for Linux, on any of the popular distros? I've looked, but can't find them.
Rosie Manning

A Neither of these programs appear to have a Linux version, but there are several good video players available for Linux, and most of them can play network streams. One of the classics is *mplayer* (**www.mplayerhq.hu**) and can play many types of file and network stream from the command line. For plain files, you can simply pass it the file name, otherwise you need to tell it the protocol to use with a full URL, such as:

```
$ mplayer rtsp://example.com/stream.mp4
```

There are plenty of command line options you can add, such as `-fs` to play the video full screen. The man page documents these options, and the keyboard shortcuts for controlling playback. Another option is *mpv* (**https://mpv.io**), which is a fork of *mplayer*. The programs are similar, as you would expect, but each has its own advantages and foibles, so try both.

If you want a GUI, there's *SMplayer* (**www.smplayer.eu**), which uses *mplayer* as its backend, although it can also use *mpv*, or the highly versatile *VLC* (**www.videolan.org/vlc**). *VLC* can be called from the command line, as with the others, by passing it a URL, but you can also open it from your desktop menu and select Open Network Stream from the media menu. *VLC* is also highly configurable from its settings menu, and has a range of plugins available if you want more than straightforward playback of either video or audio.



Several Linux video players will handle RTSP streams. VLC will even keep a list of recently viewed stream, enabling you to easily go back to them.

You can also set up desktop shortcuts with any of these, which is useful if you find yourself watching the same stream repeatedly. Use your desktop's menu to create a new launcher and set the command as above, for whichever player you choose.

Q Clone wars

I have a large folder with many files and sub-folders on my main OS SSD, which is EXT4 that I just copied to a separate SSD, also EXT4. It has the same amount of files and sub-folders, but the size of the folder is slightly different on each drive: 44,953,321,068 bytes on OS drive; 44,953,316,927 bytes on separate drive. Is this normal? I didn't modify any of the contents between the two copies.
Scott Day

A How did you make the copy? *Rsync* is the most reliable way of doing this. You can check that they have the same number of files by piping the output from *find* to *wc* (the word count program). In each directory run:

```
$ sudo find -xdev -type f | wc -l
```

You need `sudo` because some paths are only readable as root. The `-xdev` option prevent find from descending into other filesystems, like **/proc** and **/sys**. If the file counts are different, you can drill down through the tree to find out where the difference lies. A possible culprit is that you haven't copied "hidden" files – those whose name starts with a dot.

How are you getting the total file size? Most disk usage programs look at the disk space used by the files, rather than the size of the files themselves. Because it's rare for files to fill a number of blocks, the disk usage reported this way is always larger than the actual file sizes. The `--apparent` size argument for *du* deals with this.

A QUICK REFERENCE TO..FIND

The `find` command is one of the terminal's most useful tools. A simple example is:

```
$ find ~ '*.mp4'
```

This will list all of the files in your home directory with an mp4 extension, descending into any directories it encounters. The quotes are necessary when using wildcards like * because the shell will otherwise try to expand them rather than passing the search string to find as is.

There are plenty more options. `-type` finds objects of the given type: file, directory or link. This gives a useful way of finding "dangling" symlinks:

```
$ find -L ~ -type l
```

The `-L` option tells `find` to follow symbolic links. It will always identify the file that the link points to unless the link is broken, so only broken links show up as type l.

`find` is picky about the order of arguments. Those that affect its behaviour must come before the path(s) to search while search criteria come after. If you omit the path, find searches from the current directory.

Time-based searches are also possible:

```
$ find -mtime +30
```

```
$ find -newer somefile
```

The first finds everything over 30 days old, while the second returns all files newer than **somefile**.

You may want to do something with the files you find. The `exec` option runs the given command, replacing the braces with the name of each file:

```
$ find -name '*~' -mtime +30 -exec mv {} ~/.Trash/ \;
```

This finds all backup files, as created by editors, that are more than 30 days old and moves them to the Trashcan. There's a lot more `find` can do, as evidenced by the lengthy man page.

Thirdly, there's the matter of links. Most copies, such as `cp -a` or `rsync -a` will copy symbolic links as symbolic links by default, but hard links are different. For example, with *rsync* you need the `--hard-links` or `-H` option to preserve hard links, otherwise they'll be copied as separate files, increasing the space used. This isn't a default, even with the `-a` option, so you're best off making the copy with:

```
$ rsync -aH /source/ /destination/
```

Don't forget the trailing slash on each of the paths.

You said this is your OS directory. Did you copy the files while the OS was running? If so, a discrepancy is inevitable because the OS will continue to write to files, notably in **/var/log**, even if you're doing nothing else on the system. If you want to make a clone of the drive, you need to boot from a live distro on a USB stick to make the copy.

The cause of the discrepancy may well be a combination of some of these factors, and may turn out to be unimportant, but only you can decide if that is the case.

Q Browser administration

I noticed in Firefox Settings> Help>About Firefox it says "updates disabled by your system administrator". I'm the system administrator and I'd like to enable updates for *Firefox*, but don't know how.

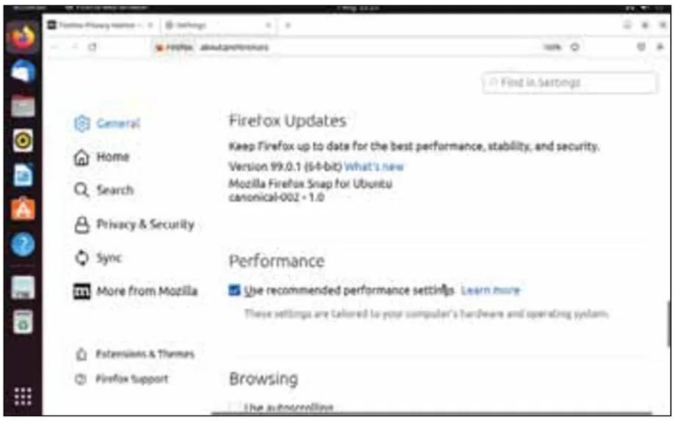
I looked up how to do this on duckduckgo but I'm a beginner and the solutions were either too complex or didn't work. What's the simplest way to enable updates?

Ben Price

A The short answer is that you need to do nothing. *Firefox*, by default, uses a Windows-style approach when it comes to updating, whereby it phones home every so often to check for updated versions, then lets you know that it's time to update. This goes against the Linux distribution model of having all software installation and updates handled by a central package manager. So it's normal for *Firefox* to show this message.

When you're running *Firefox*, you're not the system administrator but a normal user. It's only when you give the *sudo* password to the software manager that you become an administrator. So although the message is a little confusing, it's technically correct. Just let your update manager keep your system up to date and *Firefox* will be taken care of along with everything else. Incidentally, this also applies to *Google Chrome*.

As to why you are seeing this message now, that is because Linux Mint changed the way it deals with this. It used to patch *Firefox* to remove the update checking, so you would never see this message. Now it uses a policy file, which is cleaner because it means it doesn't have to change the *Firefox* program, but it means you see this warning. Mint's developers are looking at ways of removing this warning without patching.



Firefox Updates

Keep Firefox up to date for the best performance, stability, and security.

Version 99.0.1 (64-bit) What's new

Mozilla Firefox Snap for Ubuntu
canonical-002 - 1.0

Performance

☑ Use recommended performance settings Learn more

These settings are tailored to your computer's hardware and operating system.

Browsing

Linux distros take care of keeping all your software up to date and secure, so you don't need to allow browsers to phone home to check.

Q Webcam privacy

I was looking for a way to disable my webcam. I'm using Debian 11 with Gnome 3.38. I looked into Gnome's privacy settings, but they only apply to snaps, flatpaks and so on. My BIOS has an option to enable/disable the webcam, so I managed to do it with `# smbios-token-ctl`. It works just fine, but I have to reboot my laptop each time I change this setting (for example, my webcam keeps working even though I just disabled it with `# smbios-token-ctl`, until the next reboot).

Having to reboot defeats my purpose; I wanted a quick way to easily enable/ disable the webcam. Is there any way to force Linux to take the BIOS change into consideration without having to reboot ?

Louis Sullivan

A The BIOS configures the hardware before any operating system is loaded, or even a bootloader. Changing any BIOS settings are likely to only take effect at that point. However, there are other ways to disable, and then re-enable, your webcam from within Linux.

The simplest, although somewhat kludgy, approach is to simply make the video devices unreadable by any software, by changing its permissions:

```
$ sudo chmod 000 /dev/video0
```

While this works, there are a couple of drawbacks. The root user can still read it, although root would be able to undo any changes you made anyway. Also, if a program is already accessing the webcam, it may be able to continue doing so.

An alternative method is to remove the video device by unloading the module that drives it, which is almost always **uvcvideo**:

```
$ sudo rmmod -f uvcvideo
```

The `-f` option is required to force the removal even though the module is in use. This will remove the video entries from **/dev**, making the webcam disappear. That doesn't stop someone, or something, with suitable privileges from reloading the module, so you could blacklist it at the same time. A short shell script would disable the camera:

```
#!/bin/sh
rmmod -f uvcvideo
echo "blacklist uvcvideo" >/etc/
modprobe.d/nowebcam.conf
```

You could then enable it again with the following:

```
#!/bin/sh
rm /etc/modprobe.d/nowebcam.conf
modprobe uvcvideo
```

Both of these scripts would need to be run as root, or you could prefix each command in the script with `sudo`. Alternatively, use the classic "solution" of taping over the camera! LXF

GET HELP NOW !

We'd love to try and answer any questions you send to **lxf.answers@futurenet.com**, no matter what the level. We've all been stuck before, so don't be shy. However, we're only human (although many suspect Jonni is off with the guid folk), so it's important that you include as much information as you can. If something works on one distro but not another, then tell us. If you get an error message, please tell us the exact message and precisely what you did to invoke it.

If you have, or suspect, a hardware problem, let us know about the hardware. Consider installing *hardinfo* or *lshw*. These programs list the hardware on your machine, so send us their output. If you're unwilling, or unable, to install these, run the following commands in a root terminal and send us the **system.txt** file too.

```
uname -a > system.txt
lspci >> system.txt
lspci -vv >> system.txt
```

Subscriptions: for magazine issues email **help@mymagazine.co.uk**

Mailserver

WRITE TO US

Do you have a burning Linux-related issue that you want to discuss? Write to us at *Linux Format*, Future Publishing, Quay House, The Ambury, Bath, BA1 1UA or email **lxf.letters@futurenet.com**.

Fedora vs Ubuntu

I've been installing Ubuntu 22.04 and Fedora 36. (I'm using the DVD from *Linux Pro* magazine – thanks for continuing to provide a DVD! In the past when a Linux magazine quits including a DVD it has fairly quickly quit publishing the magazine, too – a subscription renewal consideration.) Fedora 36 has installed very smoothly, and Ubuntu a little less so. Ubuntu 22.04 runs fine – it was just getting it on to the partition I wanted that was difficult. Using the "Windows" button to make the Fedora 36 GNOME dock appear is feeling a very comfortable choice – and then having the full screen available.

For me, based on current experiences, the "Ubuntu Snaps under fire" concerns pointed out in issue LXF291 is just one of several reasons for me to prefer Fedora 36 now. I wish that both of your Linux magazines devoted more attention to Fedora. I have appreciated the focus *Linux Format* provides for Ubuntu and just wish that Fedora had similar amounts of coverage, so that Fedora could attract more users.

Another thing I love about Fedora 36 is that the Fedora software repositories have so many programs for amateur radio.

Toward the end of this year SDRplay plans to have new software for its hardware SDRs (Software Defined Radio), which will run not just on Windows (as is the case for its nice *SDRuno*) but on Linux, MacOS and Windows. I think that will make Fedora very popular for both amateur and shortwave radio.

Ed Scott

Which distros would you like us to do more coverage on?

Neil says...

It'll be hard for outsiders to appreciate just how expensive DVDs are to duplicate currently. The cost literally doubled at the end of 2021. So we weigh up the cost of losing subscribers and copy sales against the money we'd lose by increasing the cost of the magazine. In the UK *Linux Magazine* is £1.50 more expensive, to include its double-sided DVD of outdated software that you can download for free, is that good value?

We've started to cover Fedora a little more over the past year – and we'll likely do more in the future – and it's something we've always given coverage to, when we had that DVD thing. Good luck with the SDR – we'll have to look over this stuff again, too!

Give me access!

I could use some advice regarding replacing *Microsoft Word*. I inherited some well-made templates that make use of mergefields and rules with an *Access* database, especially incorporating Mail Merge, for my office work.

I use *Word 2016*. With Windows 10 being scheduled to be retired in 2025, I've set that as a time when I'd like to be done with Microsoft and migrate completely to open-source programs. I'm already familiar with Linux, and fond of the Debian-based distros in particular.

Can you recommend any tutorials, or perhaps would there be a paid-tier program that would offer similar functionality? Would my time be well spent learning *LibreOffice* and trying to migrate my templates and database to it? I'm willing to put some work into learning it, but reading the help manuals hasn't proven to be effective enough. I'd like your advice so I can proceed without wasting time and effort. Any advice?

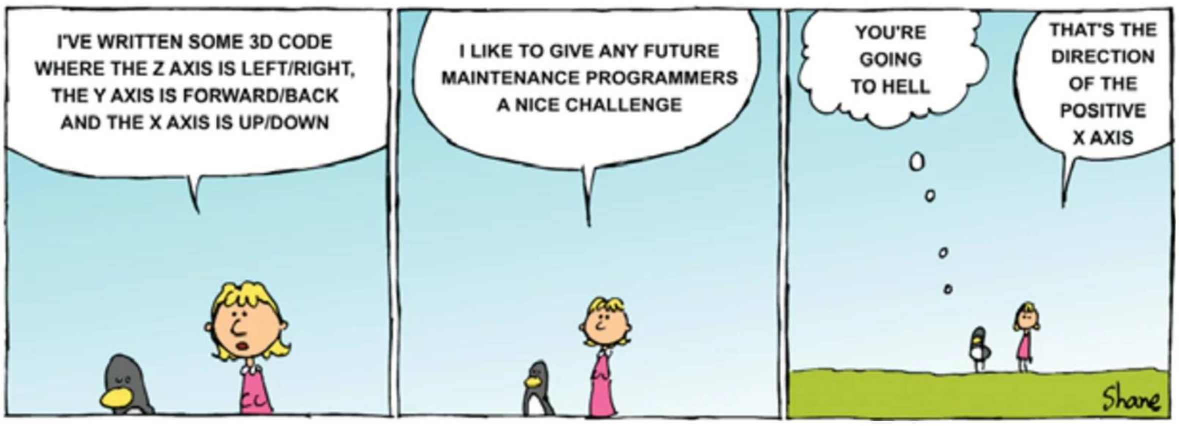
Max Stelmachowski

Neil says...

What a question! I'll say that this is way outside my area of experience, so I can't give any definitive advice, unfortunately. My vague Duckduckgoing points out that *LibreOffice Base* will use MySQL. So you'll need to export your *Access* data to SQL and take it from there with *LibreOffice Base*.

There's also this project – **www.access2base.com/access2base.html** – but it could be out of date without

Helpdex


I'VE WRITTEN SOME 3D CODE WHERE THE Z AXIS IS LEFT/RIGHT, THE Y AXIS IS FORWARD/BACK AND THE X AXIS IS UP/DOWN
I LIKE TO GIVE ANY FUTURE MAINTENANCE PROGRAMMERS A NICE CHALLENGE
YOU'RE GOING TO HELL
THAT'S THE DIRECTION OF THE POSITIVE X AXIS
Shane

New Database.odb - LibreOffice Base
Database
Tasks
Tables
Queries
Forms
Reports
Create Query in Design View...
Use Wizard to Create Query...
Create Query in SQL View...
Description

Where we're going we don't need Access!

looking more into it. Ooh, having said all that there appears to be a "LO Connector" – see this Answer (again I can't attest for this, but sounds hopeful): **https://ask.libreoffice.org/t/mdb-files-not-loading-to-base/30942/11**.

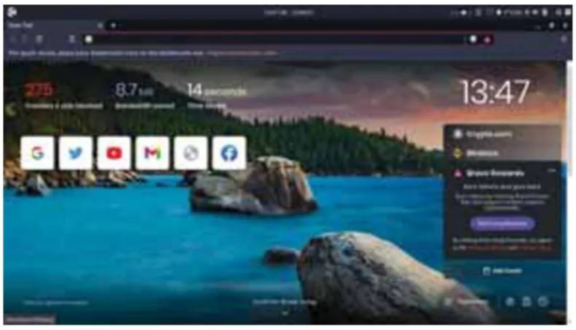
Wrong again!

I was very excited to see *Beaker-Browser* in the *Hotpicks* section after reading the "Web3 Revolution" article, only to find out that the developer quit developing the software in September 2021. Look at *Opera* and/or *Brave* for an intro to Web 3.0.

Frank Gunseor

Neil says...

That is a bit poor of us. Thanks for pointing it out, it's sort of my fault in as much as Matts mentioned it in his Web 3.0 article and I thought it might be worth writing up in *Hotpicks*. But you're totally right: *Opera* and *Brave* are obviously the most current in this area. LXF



Many recommended Brave as the best modern browser

LETTER OF THE MONTH

Illuminating!

I'm always impressed by the control of lighting during concerts, or in theatre, so I begin to wonder how much Linux is (or can be) involved? This is an area which I don't think has been covered in *Linux Format*, certainly recently.

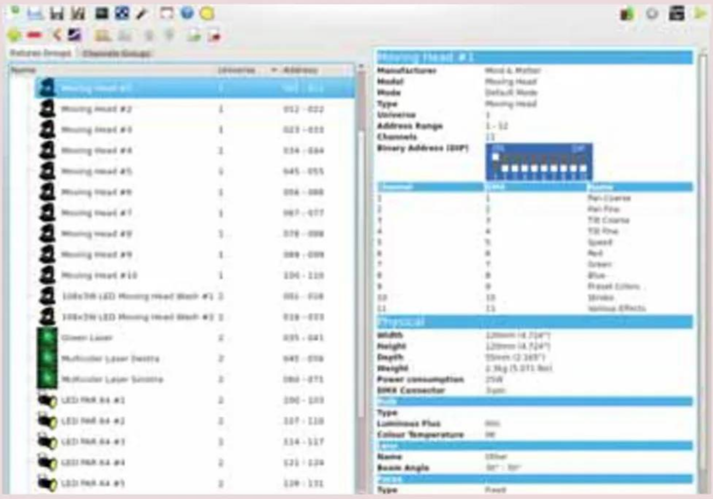
I've looked into the requirements of, say, giving a small talk that would include a couple of DMX lights for background and a projector for short videos, or photos to illustrate the subject. The best I could find is software for the presentation of lyrics in a church setting. Could *Linux Format* do better?

Andrew Stefanowski

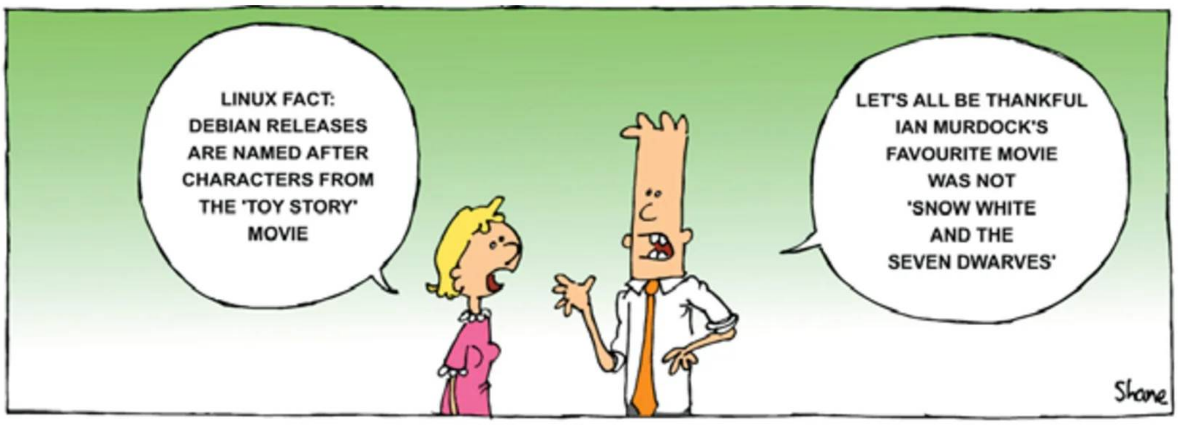
Neil says...

You mention DMX, which is the open protocol used to control professional lighting systems. This is supported on Linux and has the Open Lighting Project (**www.openlighting.org**) to help support hardware including the Rapsberry Pi, but this is just driver and the protocol support by the looks of it. What you're after is presentation software that can drive the DMX lighting, music, video and general media in one to timings. I'll openly say this is entirely outside our field of knowledge... so I've no idea if this is any good or works, but scouting around there's Q Light Controller+ at **www.qlcplus.org**. It's maintained and appears to offer the features that you're after.

Do let us know how you get on. There's certainly a host of interesting projects available based around DMX lighting.



Professional lighting experts we're not, but we'll turn our hand to anything.


LINUX FACT: DEBIAN RELEASES ARE NAMED AFTER CHARACTERS FROM THE 'TOY STORY' MOVIE
LET'S ALL BE THANKFUL IAN MURDOCK'S FAVOURITE MOVIE WAS NOT 'SNOW WHITE AND THE SEVEN DWARVES'
Shane

shane_collinge@yahoo.com

SUBSCRIBE
to *Linux Format* and get your
Laptop Bag
# worth £90!

Protect your tech while on the move with the extraordinarily spacious **STM Myth Laptop Bag**!

Product features

- Durable and water repellent
- CableReady design enables charging on the go with hidden cables
- Removable, adjustable shoulder straps with non-slip pads for extra comfort
- Impact protection isolates your laptop from the 'bump zone'
- Smart pockets, with a portable power pocket and a zippered section for your laptop and tablet for easy access
- Large 28-litre capacity – fits most 15-inch laptops

GIFT WORTH £90

Don't miss out, subscribe now!

SUBSCRIBE NOW !

www.magazinesdirect.com/lin/A68F

Call **0330 333 1113** and quote **A68F**

DON'T MISS!
Includes 5 years of *Linux User & Developer* issues

» PLUS: Exclusive access[1] to the *Linux Format* subs area!

1,000s of DRM-free PDF back issues and articles! Get **instant access** back to issue 66 (May 2005) with tutorials, interviews, features and reviews. At **linuxformat.com**

OUTSIDE THE UK?
Turn to **page 63** for more great subscriber deals!

» CHOOSE YOUR PACKAGE!

SIX-MONTHLY PRINT EDITION

SAVE! 19%

PLUS! **Only £33.75**

Six months of *Linux Format* in print by Direct Debit

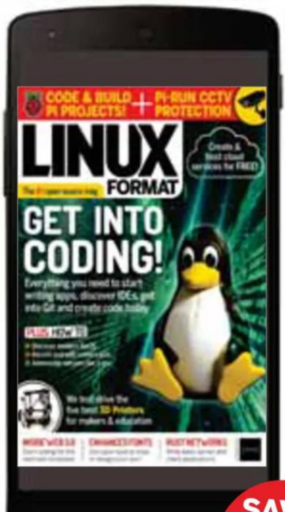
SIX-MONTHLY PRINT & DIGITAL EDITION

SAVE! 47%

PLUS! **Only £46.25**

Six months of *Linux Format* in both print and digital by Direct Debit

DIGITAL EDITION



SAVE! 39%

Only £27.50

Six issues of *Linux Format* in digital by Direct Debit

Terms and conditions: Offer closes 31 October 2022. Offer open to new UK subscribers only. Pricing is guaranteed for the first 12 months and we will notify you in advance of any price changes. Please allow up to six weeks for delivery of your first subscription issue (up to eight weeks overseas). Your gift will be delivered separately within 60 days after your first payment has cleared. Gifts only available to subscribers on the UK mainland. Gift not available with a digital subscription. The full subscription rate is for 12 months (13 issues) and includes postage and packaging. If the magazine ordered changes frequency per annum, we will honour the number of issues paid for, not the term of the subscription. For full terms and conditions, visit **www.magazinesdirect.com/terms**. For enquiries please call **+44 (0) 330 333 1113**. Lines are open Monday-Friday 9am-5pm UK time or email **help@magazinesdirect.com**. Calls to 0330 numbers will be charged at no more than a national landline call, and may be included in your phone provider's call bundle.

---

[1] Only available to www.magazinesdirect.com subscribers

Autumn Savings

FIND YOUR NEW AUTUMN READ

SPORT

TECH

MUSIC

HOBBIES

SUBSCRIBE TODAY

SEE THE ENTIRE RANGE ONLINE AT

www.magazinesdirect.com/autumn22

*TERMS AND CONDITIONS: Offer closes 30/09/2022. Offer open to new subscribers only. Savings are based on the newsstand cover price. After your trial issues, standard subscription pricing will apply. Please allow up to six weeks for delivery of your first subscription issue in the UK, and up to 8 weeks for overseas. Payment is non-refundable after the 14 day cancellation period unless exceptional circumstances apply. For enquiries, please call: +44 (0) 330 333 1113. Lines are open Monday-Friday 8:30am-7pm, Saturday 10am-3 pm UK Time (excluding Bank Holidays) or email: help@magazinesdirect.com. Calls to 0330 numbers will be charged at no more than a national landline call, and may be included in your phone provider's call bundle.

AMD Ryzen 5500

Too little, too late is what **Alan Dexter** hears a lot...

SPECS

Socket: AM4
Process: TSMC 7nm FinFET
Cores: Six
Threads: 12
Cache: 384KB L1, 3MB L2, 16MB L3
Clock: 3.6GHz (4.2GHz boost)
Mem: DDR4 3,200MHz, two-channel, no ECC, 128GB max
PCIe: v3.0 20 lanes
TDP: 65W
GPU: n/a
Cooler: Wraith Stealth

AMD released the Ryzen 5 5500, its cheapest CPU of this generation, nearly a year and a half after the first Zen 3 chips graced our PCs. Seeing as we've got Zen 4 chips due before the end of 2022, can this processor make an impact in such a tight time frame, and what's been dropped to hit that budget price tag?

The second question is much easier to answer than the first because we know exactly what the Ryzen 5 5500 does and doesn't have, compared to its peers. Like the Ryzen 5 5600X (see **LXF273**) before it, this chip is a six-core, 12-thread Zen 3 CPU, offering decent clock speeds and support for DDR4.

What it doesn't offer, though, is support for the latest PCIe 4.0 interface, being stuck as it is with PCIe 3.0. This instantly makes it a less-enticing chip if you're looking for a vaguely future-proofed purchase. It does come with the Wraith Stealth cooler though, which at least means you don't have to buy an after-market cooler to get your system up and running.

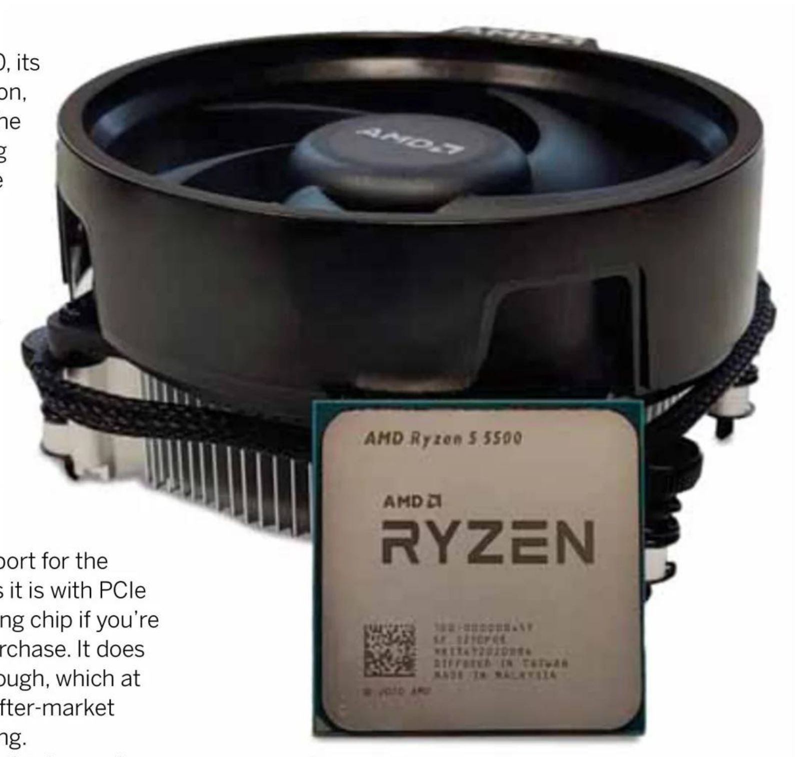
As you'd expect from a budget CPU, this has a slower clock speed than what has gone before, although not by as much as you might think. You're looking at a 3.6GHz base clock, with a boost of 4.2GHz, while the previous most-affordable chip, the Ryzen 5 5600X, has a 3.7GHz base clock and can hit turbo up to 4.6GHz.

However, the big differential is half the amount of L3 cache as the 5600 and the 5600X. And as we've seen from the release of the cache-rich Ryzen 7 5800X3D (see **LXF290**), plenty of processes love a big ol' slice of cache to kick things around in.

Testing times

While the lower frequency on its own wouldn't be a major upset, the lower cache as well makes for a less-appealing proposition. Factor in that shift to PCIe 3.0, making this less of a long-term play, and the Ryzen 5 5500 needs to deliver in the benchmarks to make it worth considering.

By way of example, the *Cinebench R23* benchmark result is neck and neck with the Ryzen 5 5600X in the multi-core test, and it isn't far off in the single-core tests either. This feat was duplicated in our x264 benchmark as well, which saw the Ryzen 5 5500 encoding video at 37fps – the same rate as the more expensive chip. This performance does come at a cost though, and in testing the Ryzen 5 5500 pulled slightly more juice than the 5600X, peaking at 78W as opposed to 76W. It also ran slightly hotter too, at 75°C as opposed to 70°C using the same AIO cooler.



We guess you do get a cooler at least.

If there is some good news here, it's that the price of this processor has dropped since it was originally launched, and can now be had for as little as £119. That's still not cheap enough to make this a straightforward recommendation though even if you're on a tight budget.

If you can't stretch to the £205 Ryzen 5 5600X or the Ryzen 5 5600 for £190, then you're left either weighing up bargain last-gen chips or Intel's Alder Lake Core i3 12100 (see **LXF292**), which can be had for around £95 (the no GPU model) and is a far more tantalising budget option. LXF

VERDICT

DEVELOPER: AMD
WEB: www.amd.com
PRICE: £119

FEATURES	6/10	EASE OF USE	9/10
PERFORMANCE	7/10	VALUE	6/10

The raw compute performance is decent for an affordable chip, but too much has been cut from the overall package to make it an attractive purchase.

» **Rating 6/10**

DragonflyBSD 6.2.2

David Rutland takes a long and confusing journey to the land before time, where things look familiar... but aren't.

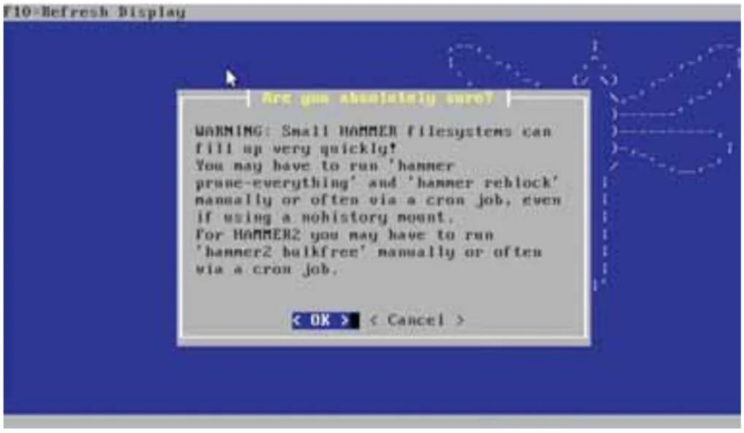
IN BRIEF

BSD-based distros are completely different to Linux-based distros and at the same time, achingly familiar. DragonflyBSD is definitely not for beginners, and the documentation while being comprehensive is not always helpful. Undocumented errors will send you to forums and YouTube. So just like Linux?

Dragonflies are glittering and gorgeous relics from another age. With a sheer size that distinguishes them from other less-dazzling insects, they zip, helicopter-like, in straight lines – defending their territory, and searching for food.

Dragonfly is also a distro, and unlike the majority of distros found in the hot, savage jungles of *Linux Format*, it isn't a Linux one. It is, instead, part of a tiny and oft-overlooked subset straight from the Jurassic era (*excuse Dave's flowery prose BSD fans – Ed*) of OS development, a BSD-based distro.

Forked from the primordial soup of FreeBSD in 2003, DragonflyBSD shares code with other BSD OSes, and boasts natively written AHCI and NVMe drivers, HAMMER, which is a "modern high-performance filesystem with built-in mirroring and historic access functionality", and according to the devs, suffers, "virtually no bottlenecks or lock contention in-kernel". Which is exactly what you'd expect from an armour-plated flying machine.



```
F10=Refresh Display

Are you absolutely sure?

WARNING: Small HAMMER filesystems can
fill up very quickly!
You may have to run 'hammer
prune-everything' and 'hammer reblock'
manually or often via a cron job, even
if using a nohistory mount.
For HAMMER2 you may have to run
'hammer2 bulkfree' manually or often
via a cron job.

< OK >  < Cancel >
```

There's no point showing you a standard Xfce desktop, so here's a snap (not the Canonical sort) of the installer instead.

Releases drop twice yearly, and are only available for 64-bit systems. If desktop environments are your thing, Dragonfly is compatible with Awesome, Cinnamon, Fluxbox, JWM, KDE, LXDE, MATE, Openbox, WMaker and Xfce.

Making this distro fly

Booting from USB, you'll be greeted with some nice ASCII art of a dragonfly, and after a few seconds, given a log-in prompt. You can either log in as root or, if you want to install Dragonfly to your system, you'll need to log in with the username installer. This is a neat approach, but requires that you actually read the text on the screen. Choose installer to make your way through the *ncurses*-style progression of options, and eventually you'll be asked to choose a filesystem. Your options are limited here, and it's unlikely that many Linux users will recognise UFS (Unix File System – a distant relative ext2), nor will they be familiar with HAMMER1 (the filesystem used by previous Dragonfly iterations), or HAMMER2, (the most recent version). Go with the default partitions and you'll be warned about the speed at which HAMMER filesystems fill up, and briefed on methods of pruning and freeing up space. There are exclamation marks, and it's all very scary to someone new to the BSD world.

Select configure system to set locale, users, passwords, home directories, network interfaces, hostnames and so on. After this, users who haven't thoroughly read the online documentation may be left even more confused. You'll need to search through the website for the term, `how to get to the desktop`. You'll find the info you need along with the unhelpful caveat that, `This topic is wildly out of date`.

You'll need to update your system, and then install XOrg, your desktop environment, a login manager, and other essential utilities such as *sudo* and a package manager, from the command prompt. This isn't always as straightforward as you'd imagine because no active repositories are configured, and you'll need to manually copy repo config files from the ISO to the disk.

In short, installing DragonflyBSD isn't intuitive, and you may have to consult resources beyond the online docs. You'll get there in the end though, and be greeted with whatever DE you chose. After a gruelling trek through the swamps of the installation, the familiar look and feel of an Xfce desktop will feel like home.

As a BSD-based system, DragonflyBSD is hugely customisable – and by necessity, you'll have done much of the customisation during the install process. It's a hard journey, and when you get there in the end, it looks like Linux, but it isn't. **LXF**

VERDICT

DEVELOPER: DragonflyBSD
WEB: www.dragonflybsd.org
LICENCE: multiple

FEATURES	**8/10**	**EASE OF USE**	**4/10**
PERFORMANCE	**7/10**	**DOCUMENTATION**	**5/10**

Dragonfly is fine, we guess. When it eventually let us into a DE, everything worked as it should. We're exhausted.

» Rating 6/10

KDE Neon 2022.08

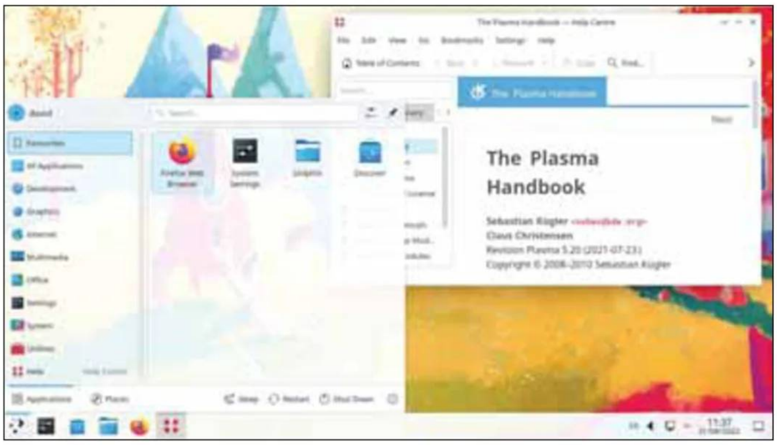
David Rutland finds that KDE is a delight to behold and regardless of whether it's a 'real' distro, it's a charming and relaxing place to be.

IN BRIEF

KDE Neon isn't an official Ubuntu Flavour. Rather, it's a KDE project with an Ubuntu core, with an emphasis on good looks and ease of use. Almost everything is extremely intuitive, and for the parts that may not be, there's excellent and accessible offline documentation.

Neon isn't a distro, we were told. It's a mere repository for KDE software, they said. But KDE Neon feels like a distro to us, and the average user will immediately recognise it as such.

It's proudly billed as the "latest and greatest of KDE community software packaged on a rock-solid base". The base in question is Ubuntu 20.04 LTS (which upon install immediately upgrades itself to 22.04), and we fail to see how the fact that KDE Neon is loaded with KDE goodness makes it any less of a valid distribution than Kubuntu, Lubuntu, Xubuntu and so on.



Did we mention that KDE Neon is very nice to look at? We didn't think it's possible for a distro to be this delightful! If you're interested, the distro's pretty capable, too.

KDE Neon comes in four varieties with various degrees of stability, but if you plan on installing on a daily driver machine, the User Edition is the one to opt for.

Booting from USB and selecting "Install" brings users straight to an uncluttered live desktop with a single Install icon, which loaded the *Calamares* installer. As always, this is a pleasure to use and a welcome relief to anyone struggling with more esoteric and DIY installation methods.

Wondrous visuals

KDE Neon's major selling point is that it's gorgeous to behold. From the softly coloured pastel wallpapers to the flat menus, the overall aesthetic is of good taste and a lack of clutter. This is thanks to the KDE Plasma 5.25 desktop environment. It's a relaxing place to hang out. Due to this graphical glamour, KDE Neon is probably more suited to modern systems than throwing on to a 20-year-old laptop you found in the cupboard.

As you might expect, the default file manager is *Dolphin*, and KDE Neon comes installed with *Firefox*, *Okular* and *VLC*. Also present is *KDE connect*: a great utility that enables file sharing, copy and paste, and even SMS control with your phone. If you have a mind to, you can use *KDE connect* to enslave your mobile device for use as a mousepad or remote control. There's no preinstalled office suite, and *GIMP* is conspicuously absent. You will, however, be able to access an Emoji Selector from the menu, which is potentially useful.

Down on the Plasma bar, you'll see an icon of a padlocked folder. This is KDE Vaults, a feature which appeared in Plasma 5.11. It enables you to quickly created password-protected encrypted folders – perfect for hiding your secrets in a hurry.

Documentation can be accessed through the Help Center entry at the bottom of the menu. It's marked with a striped lifebelt, and loads quickly, offering access to documents such as *Fundamentals*, the Plasma manual, and application manuals. We spent some time in the Help Center and found everything easy to read, easy to understand, and very comprehensive. In the unlikely event you can't find what you're looking for, the Help centre will point you towards online tutorials, user forums and the main KDE documentation site.

Because KDE Neon is based on Ubuntu, packages are installed using *apt*, either through the terminal or through *KDE Discover* – a GUI installer that's intuitive to use and makes it easy to search for the tool you need. Applications are sensibly separated into broad categories, and installing software is as simple as clicking a button and typing in your user password.

At this point, you may be wondering what is the difference between Kubuntu and KDE Neon. Both have an Ubuntu base and the Plasma desktop. It comes down to the fact that Kubuntu is an official flavour, supported by Canonical and comes with the standard Ubuntu repositories, while KDE Neon is an official KDE project, and comes with additional repositories to keep your desktop up-to-date with |KDE goodness. LXF

VERDICT

DEVELOPER: KDE
WEB: https://neon.kde.org
LICENCE: Various open-source licences

FEATURES	9/10	EASE OF USE	10/10
PERFORMANCE	9/10	DOCUMENTATION	10/10

We'd recommend KDE Neon to anyone who appreciates a good-looking desktop and has a reasonably modern PC.

» Rating 9/10

NuTyX 22.07

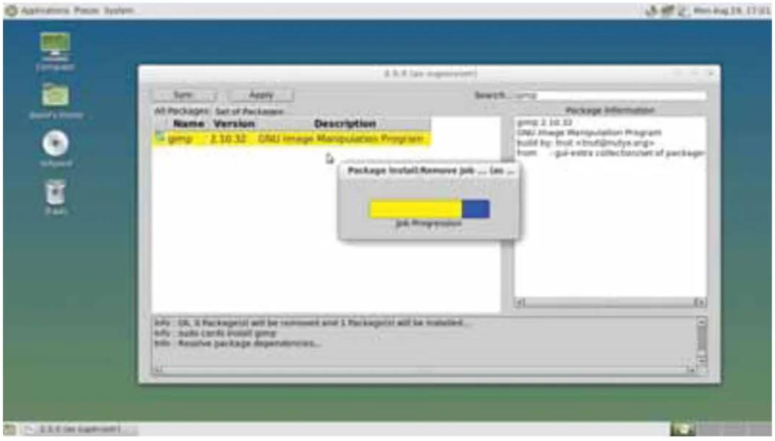
David Rutland finds this distro with its unique package management system a little less nutty than he had expected. It's certainly a bit crunchy.

IN BRIEF

NuTyx is a simple easy-to-use distro with an unusual package manager, which stays out of your way as much as possible – almost to the point of anonymity. Coming with a range of desktop environments (or none at all), NuTyX has a flavour to suit you whatever distro or other OS you're coming from, and its undemanding system requirements make it perfect for use on an old laptop.

NuTyX is an impressive French Distro with a small footprint. With the base version weighing in at a mere 715MB, it can fit on a single CD, and will run on any 32- or 64-bit system which can boast a paltry 1GB RAM. The distro is built using Linux From Scratch (see **LXF286**), and offers a unique package manager.

In addition to the base version, which offers a bare terminal interface, you can choose from desktop environments including Enlightenment, GNOME, IceWM, JWM, KDE Plasma, LXDE, MATE, Openbox, Ratpoison and Xfce.



NuTyX's FiCards is a colourful GUI frontend for its Cards package manager.

Going against the conventional grain, installation is of the keyboard-driven *ncurses* variety, and not dependent on *Calamares*. The first thing you notice is how difficult it is to find United Kingdom among the locale options, until you realise that it's the languages that are grouped alphabetically rather than the country names. United Kingdom lies a third of the way down the list, flanked by Denmark and Hong Kong. There's no keyboard detection, and it isn't immediately apparent which of the 11 QWERTY variants to select. There didn't seem to be a UK offering, so we went with US, because at least that's more-or-less familiar to us.

You'll need to name the machine and configure your network card in order to proceed, then manually set date and time, and create a user. So far, so good: you'll then log into whatever DE was bundled with your NuTyX ISO. We had chosen MATE, but you're free to choose any environment you're comfortable with. From hitting the power button on your PC, it only takes a few minutes to reach this point.

Taking stock

Unsurprisingly, the MATE desktop is stock – to the extent that there aren't even any NuTyX-themed wallpapers, just the ones which come bundled with any generic MATE install. We didn't check, but we suspect that the Xfce, LXDE, and Gnome environments are equally stock.

Installed software is stock as well, with NuTyX MATE edition boasting MATE standards including the MATE calculator, MATE dictionary, Eye of MATE image viewer, and so on. The default browser is stock *Firefox*, rather than anything weirdly unique. So far, so standard, and underneath the desktop environment you could be running any Linux distro.

Notably missing, however, are any graphics or office packages, and the easiest way to install them is to use NuTyX's very own *Cards* package manager. It's easy to use. Open a terminal and type `cards search packagename` to find a package, then `cards install packagename` to install a package.

One thing that might stand in your way is the default colour scheme in the terminal. We found that typing dark grey text on to a black background was none too easy on the eyeballs. Thankfully, MATE terminal colour schemes are easy enough to change. *Cards* can also compile software from source.

If installing software via terminal isn't for you, run *FiCards* instead. This is a GUI front-end for *Cards* that's visually and functionally similar to the *Synaptic* package manager, and can be found in the System tools menu. Just use the search bar to search for what you want, right-click, and hit Install, then Apply. You can select multiple packages to install and then let *FiCards* get on with it while you make a cup of tea. The cards sync command will update all software on the system.

We were unable to find any NuTyX documentation on the system. There's an 'About' MATE' menu entry, but for NuTyX you'll need to consult the slightly messy documentation section on the website. LXF

VERDICT

DEVELOPER: NuTyX
WEB: www.nutyx.org
LICENCE: various

FEATURES	9/10	**EASE OF USE**	8/10
PERFORMANCE	9/10	**DOCUMENTATION**	5/10

A great distro without drama that lets you get on with being productive. It's a shame about the documentation, though.

» Rating 8/10

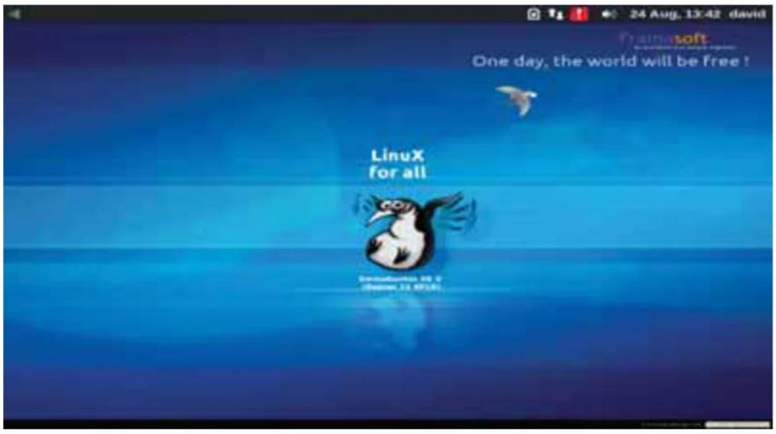
Emmabuntüs 4-1

David Rutland finds that the latest stable distro release is a little lacking in the "zig-a-zig-ah" department. Maybe it was lost in translation…

IN BRIEF

Emmabuntüs is a Debian-derived desktop distro from France, with features aimed at newcomers to Linux (who are also from France). Developers have made some interesting, although not necessarily wise, choices with installed packages and defaults. Emmabuntüs could be a daily driver for new Linux users who don't mind the odd translation error or stumbling into a Francophone charity appeal.

Based, on Debian, and with the lightweight Xfce and LXQt desktop environments available, Emmabuntüs is a lightweight distro designed with older machines in mind. It's one of only a few distros with an up-to-date edition for 32-bit machines, but there's no ARM version available. The current stable version is Emmabuntüs Debian Edition 4 1.02, although if you're willing to take a chance, an unstable version of Edition 5 is available for testing. It is also targeted at Linux beginners and migrants from other operating systems.



Emmabuntüs is a French distro, which, believe it or not, has no known connection to a certain Spice Girl with a similar name.

Installation is via the *Calamares* installer, which should be familiar to most readers. You'll set the usual options such as your name, password and locale, then reboot into the default Xfce desktop environment. You'll immediately be given the choice of switching over to the more lightweight desktop environment LXQt, and walked through various customisation options such as wallpaper, screensaver and whether or not you want the gSpeech Text to Speech engine to start alongside the system.

Just browsing

The first activity most people undertake when using a new distro is to go online. The default browser bundled with Emmabuntüs is *Falkon*. This is an open source browser from KDE, which was previously known as *QupZilla*. It does its job well, but with a couple of Emmabuntüs-specific oddities. The bookmarks bar is prepopulated with markers for the charitable work and foundations that Emmabuntüs carries out in various Francophone countries. The other inconvenience with the way Falkon is preconfigured is that the default search engine is **Lilo.org**. Lilo apparently protects both your privacy, as well as the world you live in, because money generated by your searches goes to fund social and environment projects. That's great, but the **lilo.org** homepage is an uncomfortable mixture of French and English, and despite using an adblocker, the top result with a search for the term "Linux Format" was a Bing advert for a commercial embedded Linux distro.

Changing search engine isn't easy, either. The relevant section is hidden behind a submenu, in a section of the browser preferences window marked, "other". You may find it easier to simply download another browser using the default xfce4 terminal.

If you're not familiar with the terminal then Emmabuntüs is the distro for you. It comes with *Inxi-gui* – a uncomfortable sort of GUI tool for the terminal that may be useful for newcomers to Linux. With *Inxi-gui*, you can cycle through a list of commands and their descriptions, and then launch them. Why would you want to spend time typing `du -sh` into a terminal window, when you can search for it in a list? Seriously though, a brief explanation is sometimes worth a thousand man pages.

Cairo dock offers versatility and familiarity and hovering over each icon can display subfolders containing further applications. If you're coming from the world of macOS then you'll find Emmabuntüs to be fairly intuitive.

Documentation both on the distro and on the website appears to have been carried out via Google translate rather than a native English speaker, which means you may find it difficult to navigate. Does the "assistance" tag link to a help section or to the Emmabuntüs' charitable campaign? It's the help section: the "reuse" section takes you to its charity work. This carries on in the system itself, and on the start menu you'll find an entry "No Free Softwares". Is this a manifesto? LXF

VERDICT

DEVELOPER: Emmabuntüs
WEB: https://emmabuntus.org
LICENCE: Various

FEATURES	**8/10**	**EASE OF USE**	**8/10**
PERFORMANCE	**8/10**	**DOCUMENTATION**	**7/10**

Emmabuntüs has some interesting features, and is designed for beginners. It's a shame that localisation falls short, though.

» **Rating 8/10**

Psychonauts 2

When Management said they're implementing a mindfulness policy, **Matthew Castle** didn't realise it meant entering his very psyche!

SPECS

Minimum
OS: Ubuntu 20.04 LTS 64-bit
CPU: Intel Core i3-3225, AMD Phenom II X6
Mem: 8GB
HDD: 30GB
GPU: Nvidia GeForce GTX 1050, AMD Radeon RX560, 2GB Vram

Recommended
OS: Ubuntu 22.04 LTS 64-bit
CPU: Intel Core i7 7700K, AMD Ryzen 5 1600
GPU: Nvidia GeForce GTX 1060, AMD Radeon RX 580, 6GB Vram

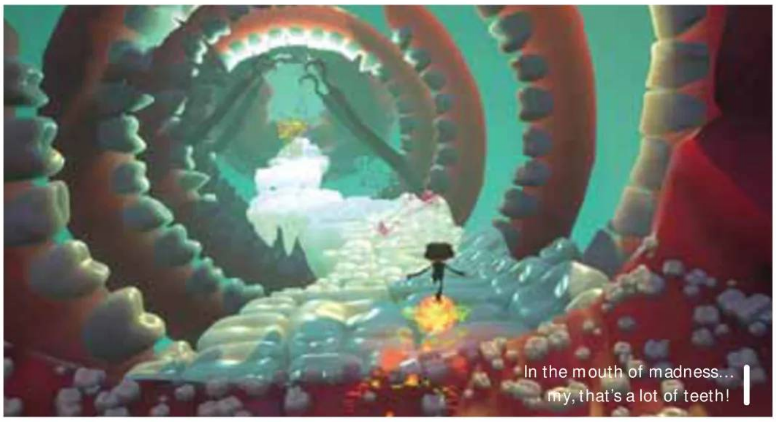
You can tell a lot about a game's soul by what it deems a 'perk'. *Fallout 3*, for example, thinks being better at murdering women is one. In *Modern Warfare* it's dropping a grenade after death like a smug ghost. In *Psychonauts 2*, Double Fine will sell you a 'beastmastery' pin badge modifier that allows Raz's psychic fist—usually used for thumping and tossing—to stroke woodland animals. You spend money to hurt fewer things. Honestly? It's a tonic.

Fifteen years have passed since the release of the original *Psychonauts*; time enough to forget how silly the game can be, and the startling places it can take us. Of course, for the game's inhabitants it's been but three days: wannabe Psychonaut Raz left Whispering Rock Psychic Summer Camp on a rescue mission – seen in the VR spin-off *Rhombus of Ruin* – and we rejoin him in its aftermath.

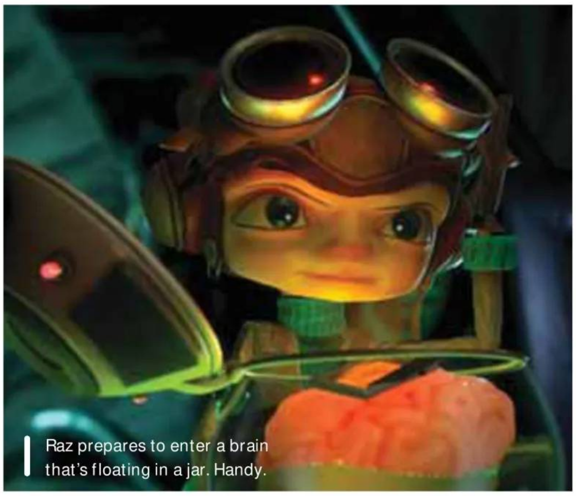
And what a start: a *Mission: Impossible*-style con to fool a villain inside his own mind, only for a dental fixation to infect the plan. With its heroes' power to enter and influence the psyche, we always thought *Psychonauts* did the *Inception* thing long before *Inception*, and this ESPionage only hammers the similarity home.

Not much has changed since 2005. Okay, it has an eye-searing prettiness that comes from an extra 16 years of HD know-how. But under levels that are variously knitted, hand-drawn or printed on X-ray film you still sense familiar cogs. It's another throwback to the 3D mascot platformers of the late 1990s. An explorable hub world siphons you into standalone levels, all festooned with enough collectibles to give even Banjo and Kazooie cold sweats.

The hub is now Psychonauts HQ, which feels like *The Incredibles* and *The Muppets* chucked in a blender: its workforce of potato-faced oddballs spit zingers, while stirring music and retro-futuristic decor sells you on the group's grand mission. After 15 hours we still regularly fast travel to the atrium just to get a hit of its swelling theme. Composer Peter McConnell does incredible work throughout the game, turning his hand to more genres than some might tackle in an entire career.



In the mouth of madness… my, that's a lot of teeth!



Raz prepares to enter a brain that's floating in a jar. Handy.

From there, Raz hops inside heads to unpick personal traumas that both infect and shape the mental landscapes. Picture Mario 64's self-contained worlds, if *Tiny-Huge Island* was a metaphor for Wario's impotence. On the one hand they serve up the comfort food of those older platformers. You hoover up hundreds of imaginary figments and earmark hidey-holes to return to with later upgrades. It's solid, standard stuff and will keep completionists happy for 20 hours. But levels are much more than obstacle courses to tidy up.

Mental narratives

Every mind tells a story. And not in the hackneyed language of audio logs and 10-foot-tall "HELP ME" graffiti. This is level design as exposition. For example, one man's familial alcoholism becomes a toxic swamp where his shame manifests as a riverbed of discarded gin bottles that you navigate before booze drowns them again. Another standout sees a bureaucrat's clinical hospital of a mind hijacked by a casino. Exploring pill-filled pachinko parlours is a visual spectacle and can be appreciated as that, but the metaphorical dimension is just as satisfying to decipher.

Crucially, there's not a tired genre trope to be found. Sick of slippy-slidey ice worlds? Try a level set on the wedding cake of a now-mourning widower. The segment that revisits a relationship in an aging, fractured mind, each memory getting jumbled with the host's stints as a barber, postie and bowling alley attendant, is lovely. It's wistful and tragic, all while conjuring disasters from human hair or a vortex of unread love letters. Weirdly, it also makes *Psychonauts 2* a good companion piece to *Yakuza: Like A Dragon* as a tale about the turmoils of an older generation that tends to hog the gaming spotlight.

This maybe paints *Psychonauts 2* as more dour than it is. This isn't one of those heart-on-sleeve misery fests you sagely nod at while secretly wishing you were playing *Peggle*. This is a game where you explore a shoe while

It'll take more than a filling to heal this molar-filled mind.

bacteria shriek about an antifungal extinction event. The best levels still flip the rules as the original's did. One mind's an archipelago explored like a miniature version of one from *The Legend of Zelda: The Wind Waker*; another, a cooking show that becomes a parody of *Overcooked* as you race to feed whooping strawberries into a blender.

Inspiration from randomness

At moments you wonder if Tim Schafer, the game's writer, is playing a kind of game design *Whose Line Is It Anyway?*, drawing random words from a hat and attempting to roll with it. What advice would Shakespeare offer Raz? How does a country's military history sound sung as a chirping Disneyland ditty? Got any gags about funicular railways? Amazingly, these are the boring questions. In the game's most audacious comic set piece you're given an entire brain to rewire, with every possible combination of thoughts resulting in a punchline. It's a literal joke factory.

We'd go further and say that as you wander the overworld and chat with fellow 'nauts it begins to resemble a LucasArts adventure. The puzzling is more physical, driven by Raz's evolving psychic abilities, but the number of weirdos you get to converse with, mining their dialogue trees for every last laugh, paints a world as alive as a Rubacava or Mêlée Island. We're head over heels for The Questionable Area, a daft hit of tourist trap Americana where you expect to bump into Sam and Max, such is the powerful oddness of its rickety attractions.

One downside to the newness is what's missing from the Whispering Rock gang. Bar Raz's (maybe) girlfriend Lili. None of the kid cadets make the trip, and their replacements, Raz's rival interns, don't have the same impact. Their shared standoffishness blurs together, and the story sidelines them after a fantastic first act.

It's a rare moment of untapped potential in a sequel that otherwise improves on its predecessor in almost every regard. Just the act of moving makes it easier to recommend. Raz is responsive, his rope and pole gymnastics are no longer a mess of button presses and the camera has your back, or whatever side you want it to have. Throw in generous checkpointing and no life counter and it's as responsive and welcoming a 3D platformer as you'll see outside of Nintendo.

Raz prepares to undertake another mental gymnastics session.

Getting the once-over from one of the staff at the Motherlobe.

Combat is likewise revamped, with an easier dodge roll and dollop of autoaim that unlocks the potential of psi powers that were too fussy in the original. Just being able to telekinetically lob scenery without wrestling with throwing arcs makes for rambunctious rumbles from the outset. And there's more enemy variety this time as flying imps, mine-laying sprites and gavel-wielding judges give individual fights a distinct flavour and ensures every ability has a chance to shine.

Given how Double Fine Productions spruces up the thrashings it's perhaps surprising how stingily it rations them out. Fights are rare in most levels – non-existent in the hub world – which makes a new upgrade system slightly superfluous. It's hard to get excited about nimbler lobs when you use them so infrequently.

Maybe it shouldn't come as a surprise that the places *Psychonauts 2* bows to convention are where Double Fine is a little more unsure of itself. This has always been a team more comfortable taking creative risks – its studio logo is a baby fired from a cannon, after all – and it's that aspect we'll celebrate. Not just the wild comic swings, but the fact that this is a mainstream video game which deals sympathetically with the messy business of being human. Strip away the screeching eggs and bacterial prophets of doom and it's a tale of regrets and the path out of them. So don that beastmastery badge: pats on the back for all involved. LXF

VERDICT

DEVELOPER: Double Fine Productions
WEB: www.doublefine.com/games/psychonauts-2
PRICE: £55

GAMEPLAY	9/10	LONGEVITY	8/10
GRAPHICS	9/10	VALUE	8/10

Improved platforming underpins a glorious return to one of gaming's most inventive and empathic universes.

» Rating 9/10

Roundup

Bibisco 2.4.0-SE » Manuskript 0.14.0
» novelWriter 1.6.2 » oStorybook 5.05.06
» Quoll Writer 3.0.0b25

Nick Peers
is 30 years a professional writer and yet only has one published short story to his name.

Creative-writing tools

Struggling to compose a story? **Nick Peers** reviews five tools that all provide more than a simple writing environment to help bring your novels to life.

HOW WE TESTED...

We took a couple of our half-formed story ideas and ran them through each program to see what they could do to help us develop them into finished pieces. We focused on examining each tool's core set of planning and organising tools, from structuring your narrative to helping you record vital information about the various elements that make up your story. We also looked at additional helper tools, from templates and questionnaires to stats helping you track your creative progress.

We also examined each applications's user interface – is it easy to navigate? How cluttered is it? The last thing you need is a tool that's too distracting to help you focus. And with distractions in mind, we also examined their core writing tools – what help do they offer you as you write, and do they provide a view that hides everything except the words you're writing, helping you to focus on the task in hand?

All you need to write your masterpiece is a vague idea and a word processor, right? Wrong. Very few people can sit down in front of their computer and bash out a lengthy story from start to finish with little or no planning. Most stories require you to build a framework providing you with an outline to follow. You'll feature characters, places and items of interest, all of which require a bit of thought before being introduced into your story.

If you're the kind of writer who relies on scribbling down notes on bits of paper as they come to you, you'll know how hard it is to bring those notes together – which is where a good creative-writing tool comes into play. These applications aim to give you all the tools you need to take your story from its initial premise to finished manuscript.

Some are more hands-on than others – actively pushing you to think about key details such as character profiles and story structure, as opposed to simply providing you with free-form tools to record what you feel is necessary.

We've picked out five creative-writing tools and put them through their paces to see which one will suit your way of working best.

Installation, upgrades

How easy is it to install and get up and running with each tool?

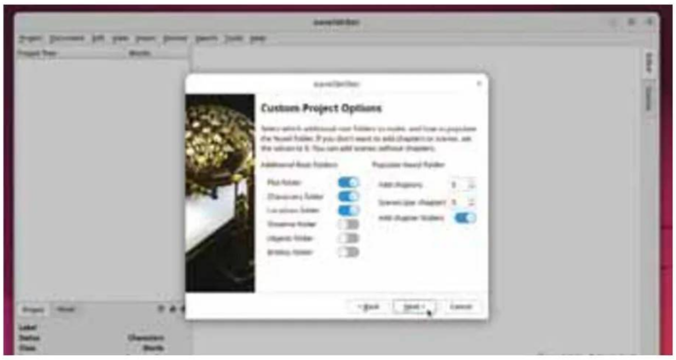
All five products are simple to install, but only three can be installed through a repo or platform that guarantees automatic updates. These include *Manuskript*, which can be installed through snap and flatpak, and *novelWriter*, which offers both stable and testing PPAs for Ubuntu users. In addition, *oStorybook* is available through Arch Linux's own repo.

(A word of warning for *Manuskript* users: after installing through Software Centre, make sure you select Permissions and switch on 'Access the NetworkManager service…'; otherwise it won't run.) All five tools provide downloadable .deb packages – some also offer .rpm and source code, like *Bibisco* and *ManuSkript*. *Quoll Writer* is currently only available in beta, while *oStoryWriter* is written in Java, so you'll need the required runtimes (`sudo apt install default-jre`) before installing the .deb package or downloading the .java version.

In terms of updates, all five packages are updated frequently with bug fixes and new features – typically at least once every six months, so be prepared to keep an eye out on the websites for new releases.

After installation, all five packages add shortcuts to the Launcher, and then lead you through a new project wizard of varying complexities. The simplest wizards are provided courtesy of *Quoll Writer* and *Bibisco*. *Quoll Writer* asks you to name your project, choose where to save it, and you're done, while *Bibisco* runs through a brief setup wizard before relying on its helper tools to steer you once your new project is in place. In contrast, both *Manuskript* and *oStorybook* give you the option of creating a structure for your story based around its type by defining word lengths or chapter and section breaks. *oStorybook* also enables you to set an optional deadline at this point.

Finally, *novelWriter* gives you complete control over how your project is pre-filled. The minimal option includes Plot, Characters and Location folders by default, while you can also add Timeline, Entities and Objects too if you plan to make use of those. Choose the Novel folder option and you'll also be prompted to set chapters and number of scenes per chapter as well.



novelWriter helps you get new projects off the ground with its customisable setup wizard. Its PPA also helps ensure you keep the program up to date.

VERDICT

BIBISCO	**7/10**	**OSTORYBOOK**	**6/10**
MANUSKRIPT	**6/10**	**QUOLL WRITER**	**6/10**
NOVELWRITER	**7/10**		

NovelWriter just wins it for its combination of PPA and configurable wizard.

User interface and navigation

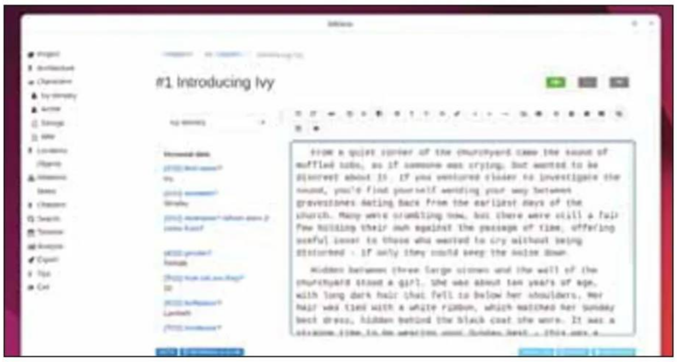
Are these tools easy to use and how accessible are their features?

Leading the pack in minimalism is *Quoll Writer* – understandable given that its toolset is less extensive than its peers. It opens with a simple two-paned interface – navigation on the left, and a multi-tabbed pane giving you easy access to multiple areas of your novel at once.

Bibisco and *novelWrite* follow the same two-paned structure, only without the tabs. Instead, *Bibsico* makes use of a back button to retrace your steps. *novelWriter* provides visible stats about the selected chapter or scene, and provides a few buttons that bring up tools like project details and writing stats in separate windows.

oStorybook's user interface looks dated as well as a little overwhelming at first glance, but everything is configurable and you can even undock items as floating windows to clear space on the main interface, and save your layout for future use.

Finally, *Manuskript* goes for a multi-paned approach – an outline on the left for navigation, a multi-tabbed window in the middle where the bulk of your editing is done, plus additional flyout tabs on the right spanning a cheat sheet, search tool, metadata and storyline – the latter of which appears in a horizontal pane beneath the other two. Unfortunately, the program tries to cram too much into one window, and even being able to strip it back fails to hide what is quite an ugly façade.



Bibisco's slick user interface is a joy to use, but there's no easy way to switch between two views quickly, unlike with other tools that offer a multi-tabbed view.

VERDICT

BIBISCO	**8/10**	**OSTORYBOOK**	**5/10**
MANUSKRIPT	**5/10**	**QUOLL WRITER**	**8/10**
NOVELWRITER	**7/10**		

Quoll Writer and Bibisco both adopt a successful, less-is-more approach.

Standout features

Find out what sets each program apart.

All of the tests we perform help us rate key aspects of these creative-writing tools, but you need to spend quality time with each to gauge just how intuitive it is, whether it suits the way you work and – ultimately – if it possesses the tools you need to help improve your writing skills and turn that long-dormant story idea into something more concrete.

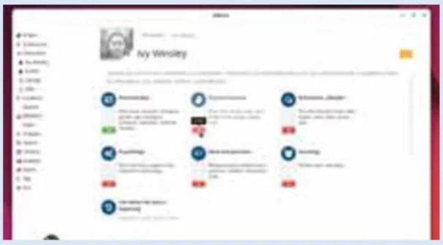
Overall, none of the five programs we tested is a complete dud. Each has its pros and cons, and each has its own unique features, too. Which one becomes your creative-writing tool of choice may come down to which one possesses a specific feature or applies it in such a way as to make it your go-to creative-writing tool going forward. If you're struggling to separate two or more tools, read on to discover what unique features each one possesses that could make it an irresistible choice for your next writing project...

Bibisco 8/10

Bibisco is the only paid-for tool in this *Roundup*. We've reviewed the Supporters Edition, but a free Community Edition exists that gives you a flavour of the main program. It supports key features like narrative strands, characters and location, but you'll need to pay to unlock the program's full potential.

The benefits of investment show: the program has the slickest user interface of those on offer. Features that stood out for us include the program's tags system, which makes it easy to link scenes and chapters to narrative strands, characters and other elements. Make the effort to do this, and you unlock its Timeline and Analysis features to help understand the pace and composition of your novel better.

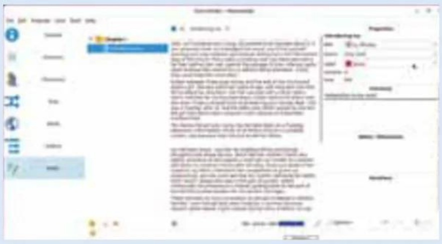
There's also a handy Relations view, which enables you to create a visual representation of the various relationships between the elements in your story.



Manuskript 7/10

There's an awful lot to *Manuskript*, but the UI feels like a barrier to getting things done as you try to work out how each section works. That's a shame, because it has some interesting features. If you're struggling with the idea of plot or character, for example, *Manuskript's* dedicated sections makes it possible to categorise threads or characters as main, secondary or minor. A Resolution Steps tab helps you determine how to bring plots to a satisfying conclusion.

Manuskript also offers a World view, which enables you to construct whatever elements you need to build a world from scratch. Viewing the blank screen is a little daunting, but click the heart button next to the Filter field and you'll see a 'Fantasy world building' section, which populates this will all manner of sections, from physical and cultural to magic systems, as well as key locations and objects.



Planning your novel's structure

How do our tools fare with helping you to organise and plot your story?

Organisational tools are the prime reason for choosing a creative-writing tool over your word processor. All five provide dedicated sections to record key characters, locations and items, from relatively simple freeform boxes (*novelWriter* and *Quoll Writer*) to more guided help that encourages you to record details you might not otherwise consider. *Bibisco* even provides you with the means of linking images to these along with additional details. Both it and *oStoryboard* score best in this department, however – offering you the choice of freeform text or detailed questionnaires to work through.

The tools also vary in offering you help with developing storylines and threads. *Quoll Writer* has a simple Ideas Board, while *novelWriter* offers a freeform notes section for developing plot ideas. *oStorybook* offers a Dedicated Plot (Actions) tab for recording basic plot devices.

Bibisco enables you to record your novel's premise in a single sentence, plus develop a synopsis using its Fabula section – a narration of events in a logical or chronological sequence. When it comes to developing the plot, Narrative Strands provides a freeform tool for recording your thoughts.

Ultimately, though, *Manuskript* offers the most help – a summary that can be a single sentence or stretch to a paragraph or a page, plus a Story line that enables you to view plots and characters with links to the scenes they appear in. Crucially, the Plots section is your chance to rank plots as main, secondary or minor. Each plot can be linked to characters, include a description and feature an outcome. A second tab (Resolution steps) helps you determine what's needed to bring that plot to its fruition.

VERDICT

BIBISCO	8/10	OSTORYBOOK	7/10
MANUSKRIPT	9/10	QUOLL WRITER	6/10
NOVELWRITER	6/10		

Both Manuskript and Bibisco provide excellent tools for managing plots as well as characters, items and locations.

novelWriter 5/10

novelWriter succeeds in balancing the need for simplicity with providing enough features for budding writers to get moving on getting their stories written. But while it holds its own with its rivals in terms of key functionality, there's very little in the way of standout unique features.

The main feature is its Build Novel Project dialog, which gives you control over your finished novel in terms of what to include in addition to your main text as well as the font and line spacing to deploy. A Build Preview button lets you see what you're getting before you click Export.

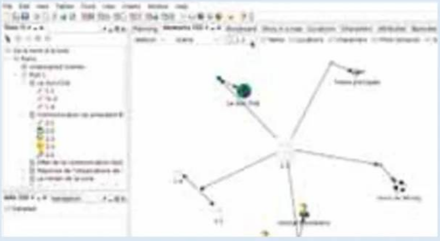
There's a handy autoreplace feature, and the distraction-free focus mode doesn't have to be in full-screen mode either. Its reliance on markdown text also means there's a handy Insert menu for not just inserting special characters, but also links to your notes in the form of tags, points of view and other references.

A Cockney in the Outback
Chapter 1

oStorybook 7/10

Many of *oStorybook's* unique features can be found under its View menu where you'll discover a Memoria, which provides you with a visual representation of the relationship between selected parts of your novel, such as characters. The interesting thing is that, unlike *Bibisco's* offering, this is generated automatically based on what you've defined elsewhere.

oStorybook is also the only application in this *Roundup* to appeal beyond simple creative-writing applications, because it includes tools for those looking to create plays or screenplays. You'll find support for splitting your stories into episodes, enabling you to generate season-long arcs, plus a Storyboard view to provide a visual breakdown of your story through its scenes. Other hidden tools include a handy overview revealing how far your project has progressed according to the status of its component parts.



Quoll Writer 7/10

The big standout feature unique to *Quoll Writer* is its Editors section. Click the people icon and you'll be prompted to register for the *Quoll Writer* Editors Service. It enables you to trade your skills as an editor for feedback from others on your story. Once registered, you'll be able to send your project to others who can highlight parts of it and make comments as part of the reviewing process. The service is free, but you can offer to pay if you're desperate for feedback, plus offer your own services for a fee.

Elsewhere, *Quoll Writer* enables you to define Research items – which are basically items with a web URL so that you can bookmark useful sites – while its Ideas Board is an area where you can store (and rate) a range of ideas. These remain hidden from the main navigation until you choose to convert them into specific elements for your story.

Ivy's Ghost
Idea Board

Keep your stories organised

How does each tool organise your story and let you alter its running order?

Writing long-form pieces is rarely as simple as starting at the beginning and writing all the way through. What a good creative-writing tool will do is provide you with tools to help you write out of sequence.

The weakest tool in this regard is *Quoll Writer*. Here, the primary form of organisation is chapters, with an option to create scene 'markers' that are inserted as pins or worked on loosely via the Ideas Board. These then appear nested under the chapter in the navigation pane; however, they contain no actual text of their own.

Other tools provide more flexibility – whether it's writing individual scenes to make up chapters or a mix of the two depending on how you want to work. These are effectively written as independent text files, which you can then reorder.

Most tools also provide you with the means of tracking your progress using a combination of status tags – here, *oStorybook* offers the widest variety. It also provides you with greater flexibility when it comes to breaking down your story, with an additional Episodes view for developing multi-part dramas, along with a handy Book View tab that provides you with a summary of each part of your story.

Equally impressive is *Bibisco* – select a chapter and you'll see its scenes, plus additional elements such as chapter-specific notes. Scenes can be rearranged via drag and drop. Take the time to tag your stories and you can view your story chronologically (complete with date and time stamps for each scene), too.

An honorary mention must go to *Manuskript's* Organise menu, which makes it possible to both split and merge existing chapters and scenes, as well as rearrange their running order.

VERDICT

BIBISCO	**9/10**	**OSTORYBOOK**	**9/10**
MANUSKRIPT	**6/10**	**QUOLL WRITER**	**6/10**
NOVELWRITER	**6/10**		

Both Bibisco and oStoryboard shine with their organisational tools to ensure that you keep on top of your developing story.

Helper and search tools

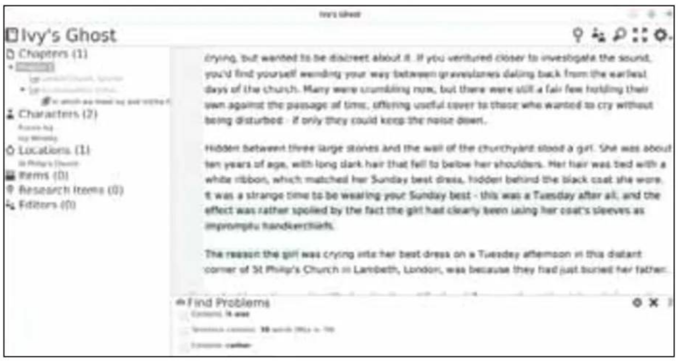
All writers need a helping hand – what tools do our programs offer?

When it comes to providing you with extra help when you need it, our creative tools shared some similar traits, but diverged widely in others. Only *Manuskript* provides any form of templates for new projects, while *novelWriter* includes a demo project and *oStoryboard* offers (French-language) demo files via its website.

Beyond basic word counts, *Bibisco* offers a complete analysis section whose offerings include chapter lengths, distribution of key elements like characters, locations and narrative strands across your novel. *Manuskript* provides a simple frequency analyser for words and phrases, while *oStorybook* provides a series of bar charts for such elements as how often characters appear to accompany its word count toolbar. *novelWriter's* Writing Statistics window helps track your progress by word count and time per session.

Ultimately, we found *Quoll Writer's* Word Counts view the most useful. A View Detail button provides word count and readability stats by chapter, session or your complete novel in bar chart form.

All five tools offer find and replace tools to varying degrees – *novelWriter* is by far the most sophisticated – but *Quoll Writer* goes above and beyond by offering two additional find-based tools. You can search for synonyms by right-clicking words within your text, while a 'find problems' tool highlights issues such as passive voice and wordy sentences. *Quoll Writer* also offers warm-up exercises in the form of a prompt and your choice of word count or time to spend on it. *oStoryboard* helps you develop ideas and characters with optional questionnaire-based assistants.



Worried that your writing style is too florid? Looking for an alternative to the word 'florid'? Quoll Writer has your back in both situations thanks to its editing tools.

VERDICT

BIBISCO	**6/10**	**OSTORYBOOK**	**7/10**
MANUSKRIPT	**6/10**	**QUOLL WRITER**	**9/10**
NOVELWRITER	**7/10**		

Quoll Writer leads the way in this section thanks to its problem finder and detailed stats tools.

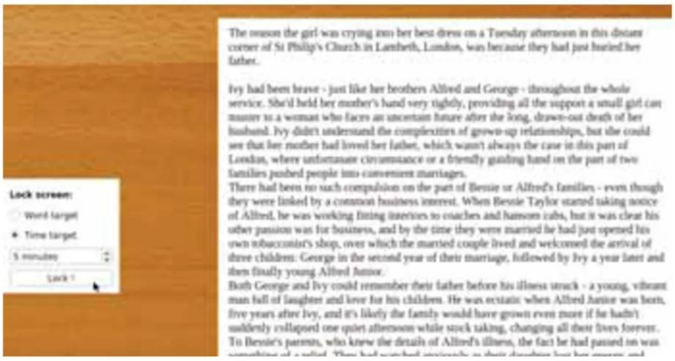
Write and export your story

How good are these tools for getting your text ready for publication?

Once you've organised all the elements of your story, you'll want to get on with writing it. While all five tools offer a built-in editor, none hold a candle to a word processor. Are there things they can do to improve your experience?

First, do they offer a distraction-free view, enabling you to focus on the writing at hand? The answer is yes for all except *oStoryboard* – its editor may be more powerful than its peers, but its Typist view merely expands the Editor to fill the screen without removing any page furniture. Of the others, *Bibisco* manages to combine some formatting tools with the ability to switch between its editor and distraction-free view. *Quoll Writer* requires you to first enter full-screen view then click a headphones icon to remove what little furniture remains; sadly its editor is the least powerful on offer, with just basic formatting available.

Both *Manuskript* and *novelWriter* support markdown text in addition to basic formatting and distraction-free mode – in *Manuskript's* case, entering this provides you with a background you'll want to customise, plus an option to lock yourself into distraction-free mode until you meet a time or word target of your choosing. *novelWriter* simply offers a 'toggle focus mode' button and leaves certain elements on-screen. Its editor does, however, score points for supporting colour-coding.



Manuskript's distraction-free view is customisable – the Linux build doesn't appear to support direct editing, but you can import themes from FocusWriter (https:// gottcode.org/ focuswriter).

All five editors come with a built-in spellchecker – only *novelWriter* offers both dynamic and manual options, but *Quoll Writer* also has its problems finder and synonym-finder tool to supplement its dynamic tool.

Finally, when it comes to exporting your work in a format suitable for publishing, *Quoll Writer* falters a little with limited support for just *MS Word* (docx) and EPUB, but the other four tools boast an impressive array of formats between them, with the likes of PDF, ODT and HTML among the formats supported.

VERDICT

BIBISCO	**8/10**	**OSTORYBOOK**	**6/10**
MANUSKRIPT	**7/10**	**QUOLL WRITER**	**6/10**
NOVELWRITER	**7/10**		

Bibisco's wide format support and distraction-free interface give it the edge.

The Verdict
Creative-writing tools

Once you've started using a creative-writing tool to help plan and plot your story, you won't revert to relying on a single *LibreOffice Writer* document for your stories ever again. All five tools in this *Roundup* will help you organise your existing notes into some coherent whole. They'll also assist you with capturing ideas as they form while providing a relatively easy means of getting your story down on paper.

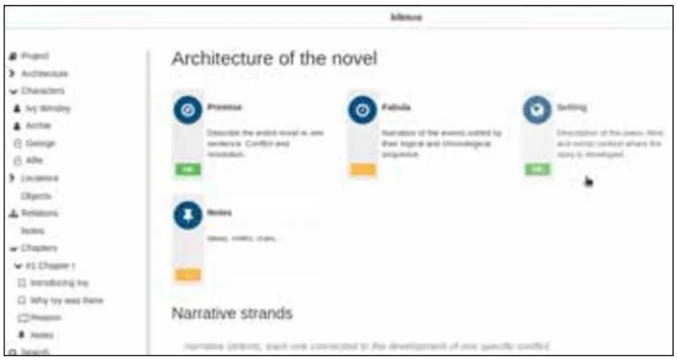
There's a reason why none of the tools here secured a 9/10 rating – even the very best lacked useful features found in rival programs. Nevertheless, we have no hesitation in making *Bibisco* our winner. While you may baulk at the asking price, there's plenty in the free Community Edition to showcase its capabilities: a slick (if not quite perfect) user interface that organises everything logically, along with key functionality. Investing in the Supporter Edition is advisable, however. You unlock *Bibisco's* full capabilities, and you may be able to steer future development. We'd love to see support for multi-tabs, for example.

We'd also like to see the developer adapt some of the best ideas from *Quoll Writer*, which was edged into second place due to its own lack of features in key areas. *Quoll's* user interface is even simpler to navigate, and we love its built-in editing tools to help improve the quality of your actual prose too, including its warm-up exercise tool to help you shake off writer's block.

oStoryboard made it into third place thanks to its powerful set of features. These, coupled with its support for a wider range of storytelling, make it worth considering, despite the use of Java and the horribly outdated user interface. *Bibisco* might like to examine its Memoria view as an alternative way of populating its Relationships tab.

Ultimately, none of the tools in this Roundup disgraced themselves. *Manuskript* could find itself pushing for top spot with a rethink of its user interface – not just the look, but structure too. *novelWriter* simply needs to add to its feature set by building on some of its rivals' better features to provide more handholding with fleshing out story elements, instead of relying on freeform text.

As things stand, however, if you're looking to finally sit down and write that novel that's been percolating in your head for years, we recommend starting your search for a suitable tool with *Bibisco*. You won't be disappointed.



1st Bibisco 8/10

Web: https://bibisco.com Licence: GPL-3.0+, Freemium (free, €23 + VAT) Version: 2.4.0-SE

The best blend of features and usability – a pleasure to use.

2nd Quoll Writer 7/10

Web: https://quollwriter.com Licence: Apache 2.0, Free Version: 3.0.0b25

Despite being in beta, this is a capable and slick tool.

3rd oStorybook 7/10

Web: https://ostorybook.tuxfamily.org Licence: GPL 3.0, Free Version: 5.05.06

A capable all-rounder in need of an interface overhaul.

4th novelWriter 6/10

Web: https://novelwriter.io Licence: GPL 3.0, Free Version: 1.6.2

A solid entry let down by a lack of standout features.

5th Manuskript 6/10

Web: www.theologeek.ch/manuskript Licence: GPL-3.0+, Free Version: 0.14.0

Some excellent tools, but confusing to navigate in key areas.

» ALSO CONSIDER

Creative-writing tools are growing in popularity, and despite the fact there are no native Linux versions of two of the pioneers in this field, you can still run both *Scrivener* (£47) and *yWriter* (free) using emulation software. *Scrivener* requires *Wine*, while *yWriter* can be run using just *mono* (`sudo apt-get install mono-complete mono-vbnc`). You'll need to download version 6 as a zip file. Extract its contents into its own folder, then double-click **yWriter6.exe** to run it. You're likely to experience glitches – visit **https://groups.google.com/g/ywriter** for help.

Those dabbling in sci-fi or fantasy should take a look at *Fantasia Archive* (**https://fantasiaarchive.com**), which is aimed firmly at world builders. The application (GPL 3.0) is freely available via either a downloadable snap or Appimage

Finally, if all you're looking for is a distraction-free editor with minimal writing tools – including spelling and grammar checkers along with basic note-taking capabilities – then check out either *Skribisto* (free GPL 3.0) or *Focus Writer* (donate to the developer at **https://gottcode.org/focuswriter**).

MULTI- BOOT MINT 21!

Got a new PC with Windows on it? **Jonni Bidwell** shows how you can get Linux Mint too, without breaking a thing.

When Windows 10 was released Microsoft stated that it would be the last version. For a time we believed the company…

Yet Windows 11 is here and if you've just bought yourself a new, TPM 2.0-bearing PC you'd be forgiven for thinking, "How the heck do I get rid of this?" Well, we don't condone killing OSes (even proprietary ones), so we'll show you the next best thing: how to dual-boot Windows 11 (or 10) with Linux.

If you've never heard of Linux before, don't worry – we're basing the tutorial on Linux Mint, one of our long-term favourite distributions that's ideal for beginners (also, bravo you for picking up a Linux magazine!). Mint is based on Ubuntu, so has a solid foundation and that enjoys five years of support from Canonical. Unlike Ubuntu though, Mint doesn't force Snap packages or dubiously configured memory pressure relief (which arbitrarily terminates important programs) upon their users.

Maybe the idea of having Linux anywhere near the vulgarity of a Windows install is upsetting to you. In that case you might be interested to learn more about multibooting various Linux distributions. Modern filesystems and partitioning schemes make all this a lot easier than it used to be, but there's still no substitute for the Logical Volume Manager (LVM). We'll show you how to harness the power of LVM to create a robust and extensible base for as many Linux distributions as you can name.

Taking precautions

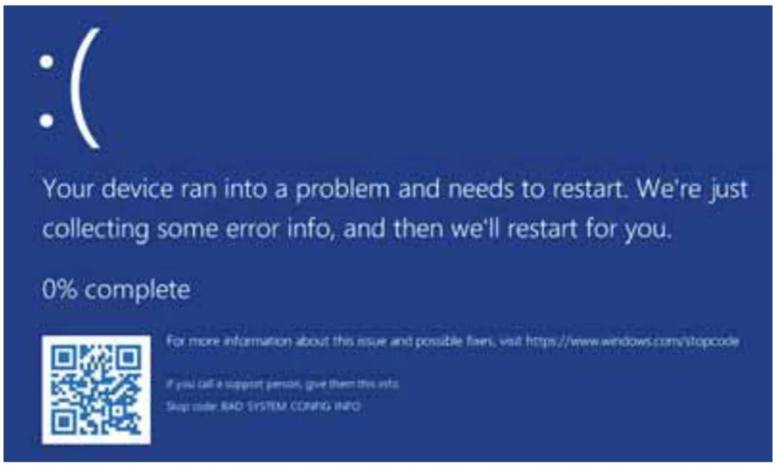
You might need Windows later so let's make sure it's not going to get upset.

Dual-booting is great. You can have a Windows install for proprietary applications (be it overpriced office suites or convoluted CAD packages) and those AAA games that don't quite work with Proton. And when you're done with those, you're just a three-fingered salute away from Linux, where you can do everything else.

But what happens when it goes wrong? Well a quick DuckDuckGo search gives one an idea: "Windows updated and broke Grub", "Windows corrupted my exotic Linux partitions", "Windows boot manager upsets my chihuahua". Okay, one of those may not be real, but dual-booting has undoubtedly caused problems in the past. We endeavour to make the experience problem-free in the present.

Before you get to installing Linux, you need to ensure that Windows is in a state which hopefully will be accepting of having a neighbour operating system. It's possible to resize Windows partitions and install on the same drive, but we've always warned against this. You can pick up a 250GB SSD for around £30 these days, which is well worth it for the reduced risk of booting to a hopeless blank screen or cryptic Windows boot manager message. If you want to dual-boot a laptop, you can use an external drive, though this might not be quite the aesthetic you're aiming for.

It's also important to turn off Fast Startup in Windows, since this may leave filesystems in an inconsistent state. So if a Linux distro were to try and mount them, errors (and possibly data loss) would result. It's also been reported that some wireless cards don't play nice with this option, since the controllers onboard awaken to find a different driver trying to command them. Accessing the Fast Startup setting is needlessly complicated in Windows. Go to Control Panel>Power Options



:(

Your device ran into a problem and needs to restart. We're just collecting some error info, and then we'll restart for you.

0% complete

For more information about this issue and possible fixes, visit https://www.windows.com/stopcode

If you call a support person, give them this info:

Stop code: BAD SYSTEM CONFIG INFO

We're hoping this feature doesn't end up with your machine getting stuck at this screen.

Confusingly, there's sometimes an unrelated Fast Boot option in the UEFI. This one is probably okay to leave on, because it aims to save time by leaving peripherals uninitialised. Either the kernel will initialise them, or you'll notice that they're not working and then sheepishly deactivate this option.

Modern editions of Windows actually don't shutdown cleanly by design. Rather they use a hibernation file to store uncommitted data (not the whole memory map as takes place with hibernation proper) and mark the filesystem as unclean before shutting down. The data is committed on recovery and then the filesystem is marked clean again. So if you were to boot to Linux in the interim, then you'd encounter an error.

Some Windows installs appear to not exhibit this behaviour, but if yours does then the solution is to explicitly disable hibernation by opening a Command Prompt as administrator and running

```
> powercfg.exe /hibernate off
```

UNHELPFUL, ESOTERIC, FIRMWARE INTERFERENCE

Windows 11 officially requires a TPM 2.0 (Trusted Platform Module) chip, UEFI firmware (as opposed to classic BIOS) and Secure Boot to be enabled. There are ways around this, but we won't discuss them here. If you have a Windows 11 machine you may well find its Secure Boot configuration has been altered to prevent you from booting Linux. Kernel security don Matthew Garrett has written extensively about this (see **https://mjg59.dreamwidth.org/60248.html**).

Historically, some Linux distros (including Debian, Fedora and Ubuntu derivatives) have had their bootloaders signed by Microsoft's third party Certificate Authority (CA) so that they worked with Secure Boot enabled. Since Windows 11 though, Microsoft has been telling OEMs not to ship systems that trust this CA by default. If you see a Secure Boot violation when trying to boot a Linux USB then it should be easy to workaround. There will be some sort of checkbox to enable a 'third party certificate' or something similar in the Firmware Options. Alternatively, just disable Secure Boot altogether, because it's only really required to install Windows 11.

A common UEFI problem (and one which firmware Fast Boot doesn't help) is that it's not straightforward to access the UEFI settings. Even if you know which key should do it, the default OS launches too quickly. If that OS is Windows, try holding Shift when you click restart (either from the Shutdown menu or lock screen), then go to Troubleshoot>Advanced Options>UEFI Firmware settings. Linux does this too (sometimes), in which case you can boot to UEFI with

```
$ sudo systemctl reboot --firmware-setup
```

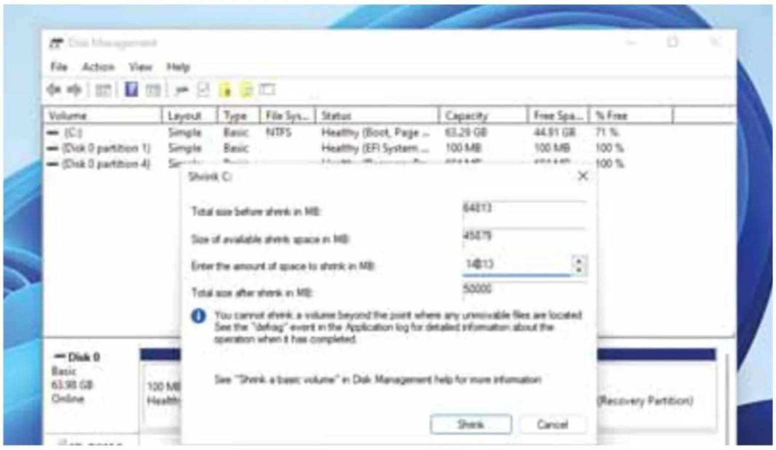
»

Installing stereo operating systems

A few more cross-checks and detours, then we can get the freshest edition of Mint installed on your machine…

First, if you're on old hardware, or hardware that's otherwise still using the old MBR (Master Boot Record, also known as MS-DOS) style partition tables, then you're strictly limited to four primary partitions. It turns out you can install Windows 11 on such a device (by addition of various Registry keys), or perhaps you have Windows 10 on an MBR drive already. In this setup Windows will occupy three primary partitions. It's possible to add one or more logical partitions, but these must exist within an 'extended' partition, which is itself counts as a primary partition.

So in an MBR setup, provided you or whoever put together your PC haven't added extra primary partitions, there's just enough room to fit Linux Mint (or whatever distro you like) on there. Of course, we just told you that's it's better to install on different drives, so MBR or no, please do still heed that advice if possible.



Volume	Layout	Type	File Sys...	Status	Capacity	Free Spa...	% Free
(C:)	Simple	Basic	NTFS	Healthy (Boot, Page ...	63.29 GB	44.91 GB	71 %
(Disk 0 partition 1)	Simple	Basic		Healthy (EFI System ...	100 MB	100 MB	100 %
(Disk 0 partition 4)							100 %

Shrink C:

Total size before shrink in MB: 64813

Size of available shrink space in MB: 45679

Enter the amount of space to shrink in MB: 14813

Total size after shrink in MB: 50000

You cannot shrink a volume beyond the point where any unmovable files are located. See the "defrag" event in the Application log for detailed information about the operation when it has completed.

See "Shrink a basic volume" in Disk Management help for more information.

Windows claims to need 65GB to install, so it gets quite upset if you try and shrink this storage amount.

Linux doesn't care about whether it's installed in a primary partition, so long as the bootloader is installed to the MBR (the most common setup). It also doesn't care about booting from partitions that don't have the 'bootable' flag set, which is a Windows artefact. And neither does it care about the 'partition type' flag. If you're starting from scratch it's best to reformat the drive with a GUID Partition Table (GPT). Then you can have partitions larger than 2TB – up to 128 of them, too.

Partition party

If you have Linux installed on an MBR drive, you may have anywhere between zero and four primary partitions. If you already have four, you can convert one to an extended partition (and then squish its contents into a logical partition using, say, *fixparts*), but this is fragile so backup first. You may already have an extended partition, in which case it's easy (if there's enough space) to add a logical partition inside there. Resizing partitions (for example, with *Gparted*) is also fragile (in the sense that a power outage mid-resize would very likely cause data loss), so again back up first.

PCs all used to use a BIOS (Basic Input Output System) to boot, but all systems built today use the modern UEFI (Universal Extensible Firmware Interface). Apart from the much-maligned Secure Boot (which we'll discuss later), UEFI makes running a dual-boot setup much easier. Instead of relying on a bootloader (or the first stage thereof) being installed in the MBR, UEFI enables multiple bootloader 'capsules' to be

THE ROAD TO PARTITION

Most Linux distributions on classic BIOS systems default to a single partition. UEFI machines require a single EFI partition, too (even if there are multiple drives). These are tidy setups, but there are other options, all of which can be easily and visually demarcated from the Mint installer. Separate partitions for **/boot** and the root filesystem used to be the norm, as well as a swap partition (which people still like to size at twice the amount of system memory, even though that guidance went out of date around the time computers started having more than 2GB of memory).

Some people still like to have a separate **/home** partition, but often not for particularly good reasons. Sharing a **/home** partition across distros is a particularly bad idea. And the belief that such a partition makes recovering data easy if the OS partition dies is generally wrong too. Unless there was physical damage to the disk (or the data was encrypted) then it's usually pretty easy to recover files from within a partition, even if the OS on there is no longer playing ball.

A swap partition (or a swap file as is the case in new versions of Ubuntu) is needed for system hibernation. If you plan on installing multiple flavours of Linux, they can all share a swap partition.

All Applications
Accessories
Graphics
Internet
Office
Sound & Video
Administration
Preferences
Places
Recent Files
Home
Desktop
Connect to...
Documents
Music
Pictures
Videos
Downloads

It's hard to say anything bad about Mint, although it's the second biggest consumer of Snaps by distro.

installed to a small FAT32 partition on the drive. This circumvents (in theory) the situation where one OS (usually Windows) steps on the toes of the other and replaces the MBR. With UEFI, it's the firmware that decides which OS to boot.

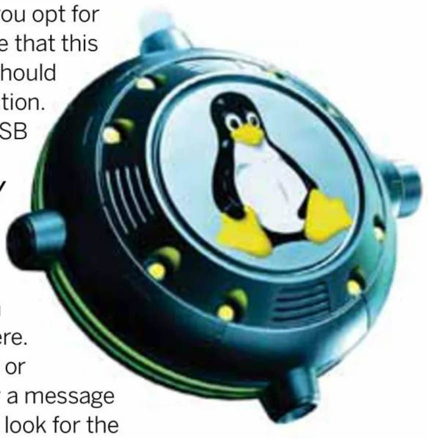
Modern disks should all be formatted with a GUID Partition Table (GPT). UEFI systems can only boot from GPT-formatted disks so we'll assume for the remainder of the feature that this is what we're running. Again, this will all be much less painless if Windows is kept on one drive and Linux (or various flavours of Linux) are installed elsewhere. If you're starting from absolute zero, the first step is to install Windows (which obviously we can't help you with).

Having discovered Windows is lacking in many ways your next task will be to install a glorious Linux-based operating system. We're going with Linux Mint, partly because we've always recommended it for beginners, partly because Mint 21 is freshly out, but mostly because we don't like people calling us Ubuntu Format. Mint is awesome, but in theory this guide should be easy to reproduce on any other distribution.

Make room for Mint

Hopefully you've got another drive to install Mint on, in which case you can get started right away. Otherwise you'll have to resize your Windows partitions and make room for Linux. This is best done from Windows itself. Hit Start and search for Disk Management. Carefully resize until there's enough room for Mint (we'd recommend 60GB if you plan on using the OS extensively, but only 15GB is needed to install). Don't worry about creating a new partition in the free space yet, the Mint installer can deal with that.

Now we can download the Linux Mint ISO and get straight to business. You can do that from Windows: just head to **https://linuxmint.com/download.php** and peruse the Mint flavours. We'll use the Cinnamon edition, but feel free to fetch either the Mate or Xfce editions if you'd rather try those (slightly lighter) desktop environments. If you opt for Linux Mint Debian Edition (LMDE) be aware that this uses a different installer, but nonetheless should still result in a successful dual-boot installation.



Now we need to write the ISO file to a USB stick. The easiest way to do that is via the cross-platform *Etcher*. Grab it from **https://balena.io**, open the ISO and wait for the etching to complete.

Next we must try and boot our freshly minted Mint medium. This will be easier on some machines than others, so do persevere. There will be a shortcut key (commonly F2 or Delete) to enter the UEFI setup, and ideally a message will pop up telling you what this is. Once in, look for the boot priorities and promote the USB device to the top. Most systems also respond to a different shortcut key (commonly F10 or F12), which summons a handy one-time boot menu. Some systems offer to boot it in either Classic BIOS (sometimes called CSM) or UEFI mode. Choose whichever your primary OS was installed (hopefully UEFI), otherwise you'll lose access to Windows. Hopefully in a matter of seconds you'll be looking at Mint's Welcome screen.

If not, don't fret. The box on the previous page has some tips for obstreperous UEFI interfaces. If you get past any UEFI screens but then encounter an error (or a blank screen and no sign of life), then there may have been a problem writing the USB. It's also possible (though unlikely) the ISO download was corrupted. You might also ask on the Mint forums, but without extra information such problems rarely attract solutions.

Let's assume you've made it to the live environment. You could jump straight into the installer, but at this point it's wise to connect to your wireless network and check any other hardware is working as it should. You should also take a moment to enjoy the wonders of Cinnamon (or whichever desktop you chose). Now start the installer, select your language, tick the multimedia codecs box and we're almost there. Mint will detect Windows Boot Manager and offer to install alongside it. However, if you've got another drive you'll want to choose 'Something else'. Turn the page to see exactly what that thing is.

"Nothing is certain, except for system updates, death and taxes," to paraphrase Ben Franklin.

Update Manager
File
Edit
View
Help
Clear
Select All
Refresh
Install Updates
Computer
Home
Type
Upgrade
Name
Description
Packages
Cancel

Downloading Package Files

Downloading Package Files

Download rate: 1173 kB/s - 1min 37s remaining

▾ Show individual files

Status	Size	Package	URI
Done	582 kB	libsdl2-2.0-0	http://ar
Done	201 kB	libsnmp-base	http://ar
Done	1070 kB	libsnmp40	http://ar
Done	1862 kB	libxatracker2	http://ar
Done	1083 kB	lintian	http://ar
Done	238 MB	linux-firmware	http://ar
Done	22.7 MB	linux-modules-5.15.0-47-generic	http://ar
Queued	11.4 MB	linux-image-5.15.0-47-generic	http://ar

Tweaking the Mint

Finish the Mint installation, enjoy a fresh view of your Windows files and learn more about UEFI.

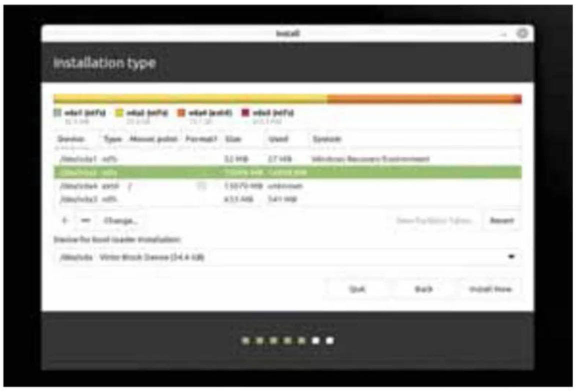
The "Installation type" dialog that pops up when you click "Something else" is a little intimidating at first glance. You should see a map of your Windows drive and a colour-coded list of its partitions, as well as any free space in between them. If you've got another drive it'll likely appear at the bottom of the list, so scroll down to it. If it's unformatted you'll need to click New Partition Table and accept the warning that all current partitions will be removed. Don't use this option on your Windows drive unless you don't want to use it ever again! (*Is that a bad thing?–Ed*) You'll now be able to select the free space in the new drive and create partitions for Linux Mint by clicking '+'.

The simplest setup (and hence what we'll use) is to have a single partition occupying the whole drive. Mint by default will select the Ext4 filesystem, and so should you, unless you have burning reasons for doing otherwise. In the mount point box enter a `/` so that the installer knows that this will be the root filesystem. Ensure the 'Device for boot loader installation' setting remains on your first (Windows) drive, otherwise a world somewhere in between pain and confusion awaits. This option is in fact a little misleading on UEFI installations, since the bootloader is installed not to the device and on top of the incumbent one, but to the EFI partition on the device, alongside the incumbent. So there isn't (all going to plan) any risk of losing access to either OS.

That being said you'll never know if things are borked until you reboot. So assuming the installation completed without error message, bravely hit Restart Now and see what happens. If the worst happens, and you end up with a rather impotent-looking Grub prompt, don't worry, type `fwsetup` and you'll be taken to the UEFI settings, where hopefully you'll be able to fix things. It may be as simple as nudging the UEFI to load the correct capsule. Confusingly Mint (and most Ubuntu derivatives) labels itself as `ubuntu` in the UEFI, which is slightly annoying when it comes to multibooting Ubuntu derivatives.

Curiouser and curiouser

We said we weren't going to discuss classic BIOS/MBR setups anymore, but while playing around with various combinations and permutations of such things we did happen upon a curiosity. If you can trick Windows into installing to an MBR disk, then it'll create three primary partitions, as expected. If you then attempt to install Mint on to this disk, you'll see an error about a missing EFI partition. This we found bizarre, since common sense says all Mint should need was a new primary partition, which there was just space for. Why this



What no EFI partition? Even Windows 11 can (under duress, it must be said) install to a MBR drive.

KEEPING THINGS SIMPLE

"The simplest setup is to have a single partition occupying the whole drive. Mint by default will select the Ext4 filesystem, and so should you."

» EFING AROUND…

While the EFI capsules that kick our OSes into life are stored in the EFI partition (usually mounted at **/boot/efi/** on Linux) the data about which order to attempt loading them is stored in non-volatile memory (NVRAM) deep in the firmware. This and other data is collectively referred to as the EFI variables, or efivars for short. In general you shouldn't be messing with these, but UEFI is a complicated standard (most implementations run to millions of lines of code) and occasionally there are bugs. If, for example, you find you're unable to change the boot order from the firmware interface, then you might be able to use *efibootmgr* to fix it.

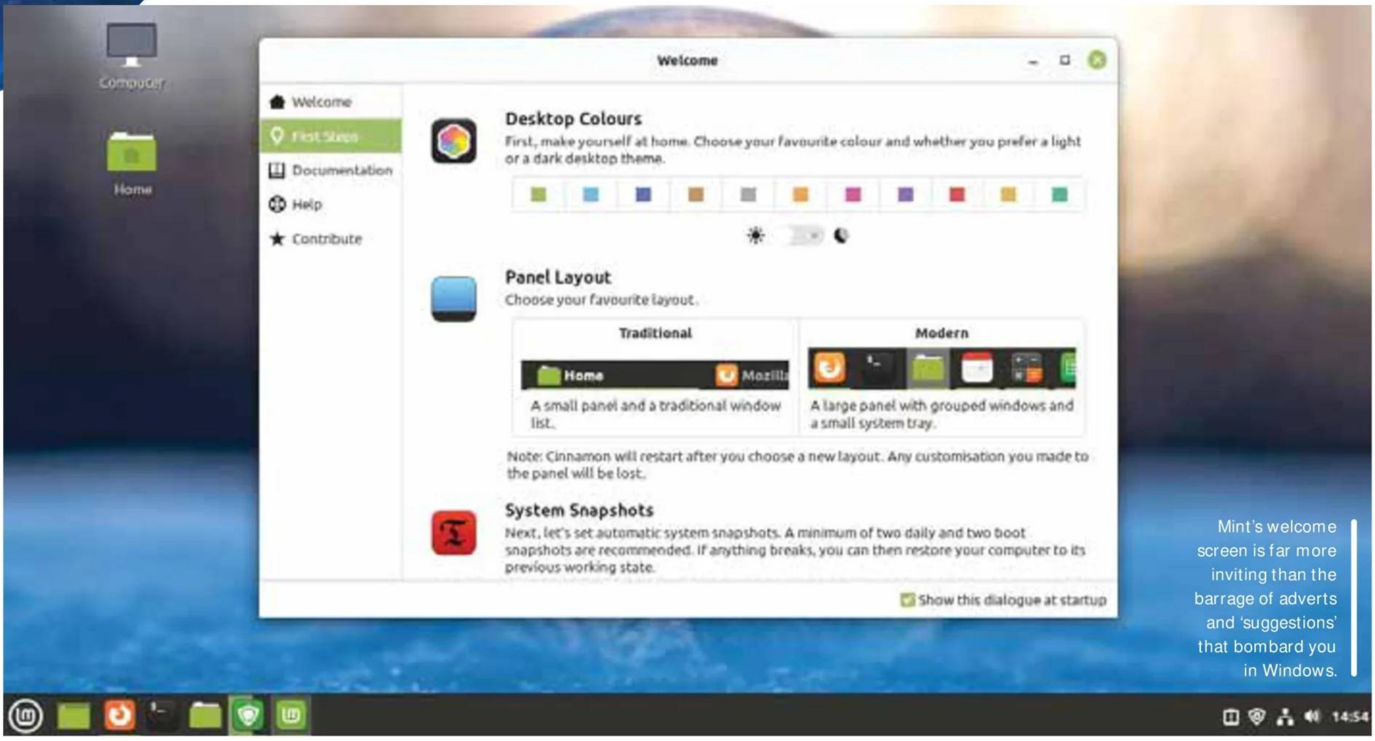
Running the command on its own might output something like:

```
BootCurrent: 0005
Timeout: 3 seconds
BootOrder: 0001,0005,0004,0002,0000,0003
Boot0000* UiApp
Boot0001* UEFI DVD-ROM QM00005
Boot0002* UEFI HARDDISK QM00001
Boot0003* EFI Internal Shell
Boot0004* Windows Boot Manager
Boot0005* ubuntu
```

To make Ubuntu boot first, preserving the rest of the order, run:

```
$ sudo efibootmgr -o 0005,0001,0004,0002,0000,0003
```

There isn't really a Windows equivalent of *efibootmgr* to, but there is *BCDEdit* for editing Windows Boot Configuration Database. This deals only with Windows Boot Manager…



Desktop Colours

First, make yourself at home. Choose your favourite colour and whether you prefer a light or a dark desktop theme.

Panel Layout

Choose your favourite layout.

A small panel and a traditional window list.

A large panel with grouped windows and a small system tray.

Note: Cinnamon will restart after you choose a new layout. Any customisation you made to the panel will be lost.

System Snapshots

Next, let's set automatic system snapshots. A minimum of two daily and two boot snapshots are recommended. If anything breaks, you can then restore your computer to its previous working state.

☑ Show this dialogue at startup

Mint's welcome screen is far more inviting than the barrage of adverts and 'suggestions' that bombard you in Windows.

happened we can't say. It could be a bug with the Mint installer, it could be a quirk of virtualisation, or it could just be us losing track of our boot modes. At any rate, if any readers have Windows installed in a Classic BIOS setup, do let us know if Mint (or potentially anything of Ubuntu lineage) displays this odd behaviour.

Adding some Mint

It's worth following the guidance in the Mint welcome wizard, especially regarding System Snapshots. These won't prevent problems relating to dual-boot, but they might help you recover from others. Now you're free to set up Mint however you like, such as colours and wallpaper. You'll be coerced to update the *Update Manager*, which will then present you with some updates (don't worry it won't take long, unlike on that other OS). Hopefully you'll want to add an ad-blocker to *Firefox* (de-snapped and deb-packaged for your convenience by Clem and co). Maybe you'll even want to connect some Online Accounts. These are all bread and butter tasks that are well documented elsewhere.

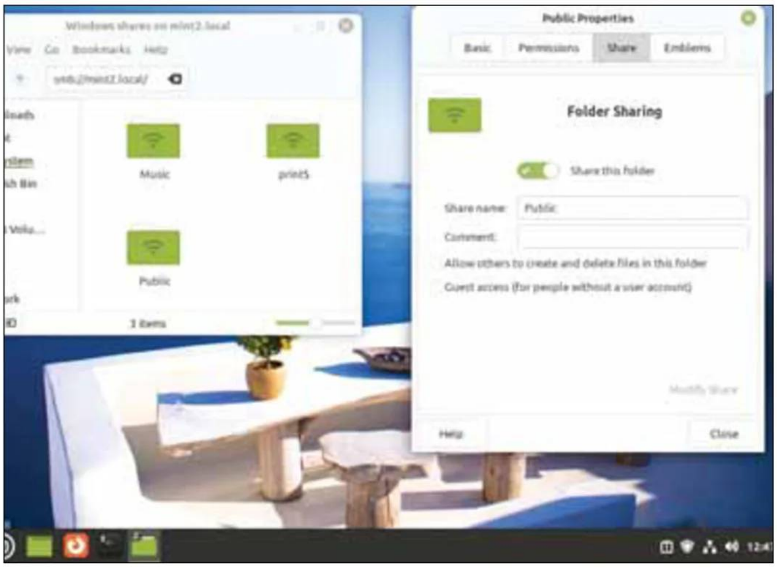
There's not really much else to do that's particular to a dual-boot setup. Naturally you should check that your Windows partition is visible from the *Nemo* file manager. It should show up in the Devices section, and unlike days past you should have read/write access without having to do a thing. For convenience you may want to make a bookmark to your home folder on the Windows drive. They all live in the **/users** directory, so find your user there, right-click it and select "Add to favourites". That will make it ever so slightly more convenient to copy files between OSes, because without additional software Windows won't be able to view Linux partitions. Mint ought to be able to see any other Windows machines on your network and access them without issue from *Nemo's* Network > Windows Network section.

If you're interested in sharing directories from Mint to other Windows machines you'll need to first install *Samba*. Do that by opening up a terminal and running:

```
$ sudo apt install samba
```

This will automatically add your user to the `sambashare` group, which will enable you to access the Sharing Options dialog when you right-click a folder in *Nemo*. You'll be helpfully reminded of any permission issues that might impede others accessing the folder before you click Create Share.

A common pitfall here is that the parent directory of whatever your sharing needs to have its executable bit set for everyone (not a security risk). So, for example, if you try and share the (seemingly created for the task) **Public/** folder in your home directory, it won't work (at first). To make it work you'll need to go to your home directory, right-click somewhere in the empty space and select Properties. Now go to the Permissions tab and set the third Folder access drop-down (which governs users outside of the directories group) to Access Files.



Share this folder

Share name: Public

Comment:

Allow others to create and delete files in this folder

Guest access (for people without a user account)

Once you have Samba installed you can easily share files from your Mint install to the rogues on your Windows network.

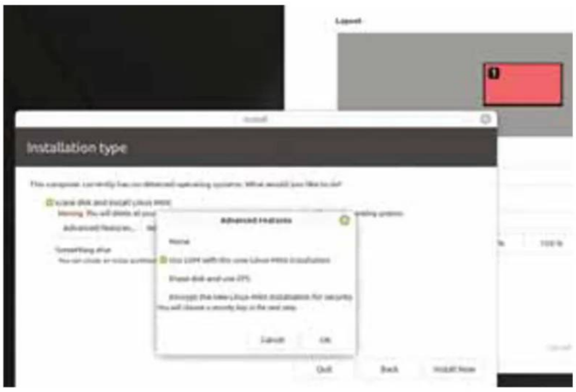
»

Logical Volume Manager

Multiboot many distros with aplomb through the magick of LVM.

Besides dual-booting Windows, you might be interested in dual-booting another Linux distribution, or even multibooting several distros. It's fine to have multiple Linux distros sharing the same drive. There are a few things to watch out for, but in general the penguins play nicer together compared to Windows. Thanks to flexible modern filesystems, you can even resize partitions while they're mounted, although unless the need is urgent it's always going to be safer to unmount first.

Thanks to GPT partitioning there's no need to worry about fitting in another distro (unless you've gone a bit wobbly and installed about 126 already). And thanks to UEFI booting you'll never lose the ability to boot either distro. In fact, most distros will enable you to boot any sibling OSes from their boot menus. So the orthodox procedure for adding a new distro is as simple as the three-step walkthrough below suggests. It's often even easier, since most distros make it possible to do any partition shuffling directly from the installer. This should work just fine with the Windows-Mint arrangement we created on the previous pages too, especially if you installed Mint on another drive. In this case one should definitely put the new distro on the Mint drive, rather than the grumpy Windows one.



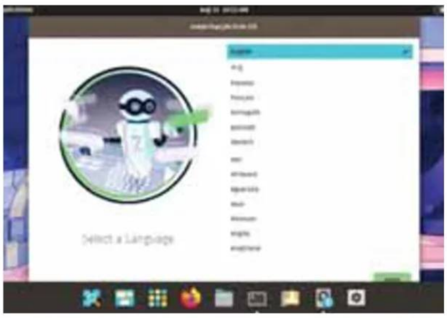
The Advanced Features button opens the door to all kinds of geek-worthy gubbins.

Take the logical view

To finish off, we'll look at a more advanced way to accommodate multiple distros: the Logical Volume Manager (LVM). Before all these fancy filesystems came about it used to be tricky to resize or move partitions around. So doing what we do in the walkthrough below wouldn't have been an option a decade ago, or if it was it would have been a risky one. To work around this, people with peculiar partitioning proclivities resorted to logical volume management. Rather like the extended partitioning on MBR disks, LVM abstracts physical partitions away, and deals purely in logical ones. Logical partitions may be spread (or mirrored) across multiple disks and can safely be manipulated on the fly.

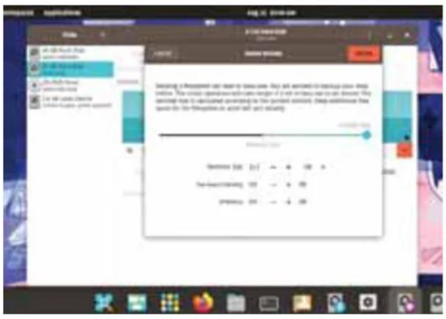
Back in the day, if you wanted to set up your own logically managed volumes you'd have had to brave all kinds of command-line tools and space calculations. These days it can all be done graphically. Also, when in the past you'd have had to make special provisions for a **/boot** partition (since bootloaders of the day couldn't access LVM partitions), today that's no longer a concern. If you looked in the Advanced Options menu in the Mint installer you might have even noticed the option to use LVM with the new install.

SIMPLE LINUX DUAL-BOOT



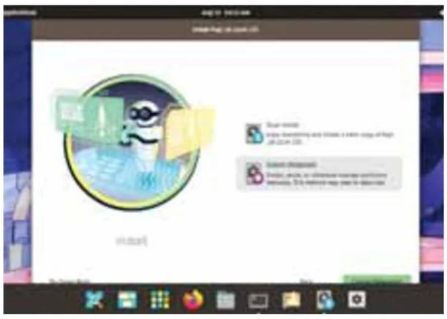
1 Boot a new distro

Let's say you already have Mint installed. First boot the distro you want to install alongside. We'll go with Pop!_OS, another of our favourite Ubuntu derivatives. This, too, includes Gnome Disks, so skip through the welcome screens and start it. Select the Mint partition, click the cogs beneath and select Resize…



2 Make room

Shrink the Mint filesystem so the new distro can be accommodated. Pop!_OS needs at least 10GB to install and probably a little more to grow. It also needs a larger than normal EFI partition, so that may need resizing, too. Assuming there are no scary errors, close *Disks* after the resize is complete and start the Pop installer.



3 Install the new distro

Select the Custom (Advanced) option and tell Pop!_OS to create an ext4 filesystem in the newly created free space. You'll also need to tell it to use the EFI partition (which is on the same drive as Windows if you've followed from the beginning) at **/boot/efi**. Click the install button and soon you'll have two Linuxes.

Before we go further we'll introduce three bits of nomenclature which are the cornerstones of LVM:

- **Volume Group (VG)** – this is the highest stratification of LVM structures. It's a collection of physical and/or logical volumes. More often than not there's never any need for more than a single VG.
- **Physical Volume (PV)** – a block device (for example, a regular partition) that will hold our logical volumes.
- **Logical Volume (LV)** – a placeholder for a filesystem. Rather like partitions, except that a LV may span several PVs, and the space inside one needn't be contiguous. This obviates most situations where partitions need to be moved. Instead, if we need more space then we can add it from somewhere else in the VG.

We're not sure if we'd recommend installing LVM alongside Windows at this stage. It worked in our testing, but there no critical data was at risk. Our main concern was that a future Windows update may just wipe everything out. Or just see the VG's territory as space that could be commandeered by Microsoft's Storage Spaces, or something equally abhorrent. But if you're planning a Linux-only install, then it's a great idea.

Working with a single partition

Let's say you've got a nice fresh disk on to which you install Mint using the LVM option using a single partition as a LV (this is what you get if you choose the usual "Erase disk and install" option). You'll end up with a VG called **vgmint** with two LVs named **root** and **swap_1**. Presumably whoever coded that figured more swap partitions might be needed in future. We can start to get some information about our array by running:

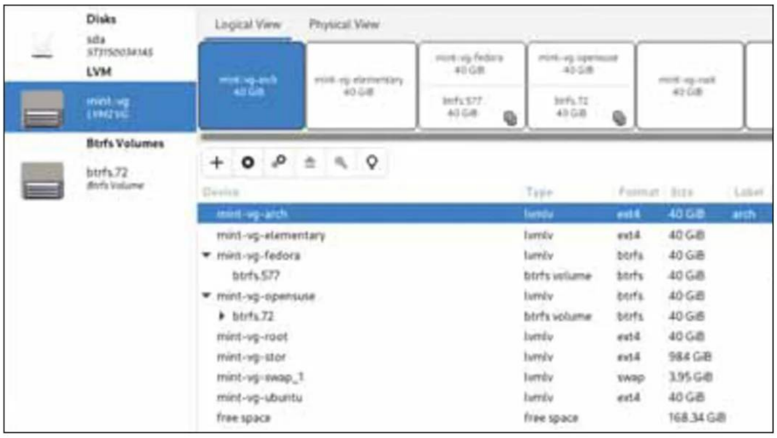
```
$ sudo lvmdiskscan
```

You'll see a normal EFI partition (if you installed in that mode) and a single LVM physical volume. The old way of administering LVM was using the commands such as `pvcreate`, `lvextend` and the like. But it's possibly to do everything graphically thanks to Fedora's marvellous *Blivet-GUI* application. This started out as an alternative partitioning utility for Fedora's installer in 2015. Shockingly it still hasn't made it to the Ubuntu repos, but it's easy to install from the OpenSUSE Build Service. Head over to **https://build.opensuse.org** and search for **blivet-gui**. Scroll down to Ubuntu 22.04 and click Show Community Packages. There's one called **home:vtrefny** that you'll want to click. Click Download Package in the top-right, select Ubuntu (again) and click "Grab binary packages directly". There's a .deb file for Ubuntu 22.04, which you should duly grab.

If you try and install that immediately, you'll see an error that **python-blivet** isn't installed. Our quest for modernity may have landed us in dependency hell. Let's avoid that by going back to the *Blivet-GUI* page and instead clicking "Add repository and install manually". Copy and paste the commands you see there and in no time you'll have achieved something rather circuitous: using OpenSUSE's build service to fetch an Ubuntu package of a Fedora program and then installing it on Linux Mint. Keep the names of these distros in your mind somewhere, because they'll reappear soon.

Create new device
Cancel
OK

Our virtual machine install didn't really have enough space to fit another distro in, but you might.



Device	Type	Format	Size	Label
mint-vg-arch	lvmlv	ext4	40 GiB	arch
mint-vg-elementary	lvmlv	ext4	40 GiB	
▾ mint-vg-fedora	lvmlv	btrfs	40 GiB	
btrfs.577	btrfs volume	btrfs	40 GiB	
▾ mint-vg-opensuse	lvmlv	btrfs	40 GiB	
▸ btrfs.72	btrfs volume	btrfs	40 GiB	
mint-vg-root	lvmlv	ext4	40 GiB	
mint-vg-stor	lvmlv	ext4	984 GiB	
mint-vg-swap_1	lvmlv	swap	3.95 GiB	
mint-vg-ubuntu	lvmlv	ext4	40 GiB	
free space	free space		168.34 GiB	

Blivet GUI as it looks in the Fedora installer, and seven distros, all working!

Blivet-GUI should now be available in the applications menu. Start it up (you'll be asked for your password) and you'll be presented with a "Logical View" of your disks. This won't be a particularly interesting view at this stage, so instead click our Volume Group, **vgmint** in the left-hand column. You'll be able to see the two LVs set up by the Mint installer. If you want to add another LV right-click inside the free space and select New. If there's enough free space, we can install a whole new distro in there. If there's not, you'll need to add more, either by adding more storage and connecting it as a Physical Volume, or by resizing the filesystem and then the LV on which Mint is installed.

PARTITIONING, OLD-SKOOL STYLE

> "The old way of administering LVM was using the commands such as 'pvcreate', 'lvextend' and the like."

Currently *Blivet-GUI* doesn't support doing this on the fly, but it's included on the Fedora live medium (and can be installed in the Mint live environment exactly as we did above) so there are a few options for working around this. Once you have free space in the VG, you can install another distro in there almost exactly as normal. The illustration shows how Pop!_OS happily will pick up any volume groups and install into them. What it doesn't show is that it (or rather *Gparted* that comes with the live environment) doesn't enable you to create new LVs and filesystems in free space. That is once again a job for *Blivet-GUI*, and once you've mastered that, you too can have seven (or more) logically managed volumes, all coexisting peacefully. LXF

DISCOVER OUR GREAT BOOKAZINES

From travel and history to gaming and photography, you're certain to find something you're passionate about

Follow us on Instagram @futurebookazines

FUTURE

www.magazinesdirect.com

Magazines, back issues & bookazines.

3Jw8GGqdiRsHphQEjEgfZw==

Les Pounder works with groups such as the Raspberry Pi Foundation to help boost people's maker skills.

» RIP OLLY CLARK

Sad news reached the UK Linux and Raspberry Pi community that Oliver Clark, Olly to his friends, sadly passed away. Olly was a titan of the community. He was there at the grass roots, helping everyone around him. In the early days of the Raspberry Pi he was there for the first Raspberry Jam in Manchester, helping children and adults get to grips with the new tech. It was also where Olly casually sat on a tabletop, which caused the table to flip and launch an LCD monitor into the air. Thankfully it landed without any issue. We also can't forget the "dongle-chain" that saw HDMI converted to VGA using every dongle known to man, only to realise that his laptop had VGA after all.

Olly was a remarkable man. He knew his stuff with Linux and free software and used both to build his business. Many will know Olly via Oggcamp, the Linux and free culture unconference. Olly was part of the legendary crew, a crew that I had the honour of being chief to. Olly was hands-on, moving kit, answering attendee questions and so on, but his specialist skill was audio/video. Along with Dan Lynch and the crew, Olly made sure that Oggcamp worked and he made many friends at the event.

While we mourn the loss of a friend, we're thankful that we knew Olly and that he was part of our lives. His legacy will live on in the lives of those he helped, and the memories of those who knew him. Rest in peace Olly.

RS Group loses its 10-year Pi licence

RS Components had been a manufacturer and distributor of Raspberry Pi since its launch in 2012.

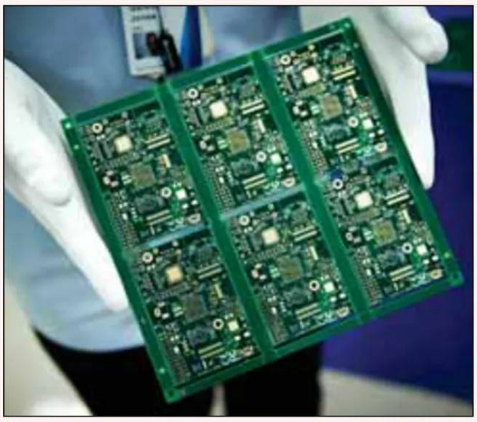
Back in 2012 there were two main manufacturers and suppliers of Raspberry Pi boards: Element14 (part of the Premier Farnell network) and RS Group (RS Components). It was these two websites that took the full brunt of the eager masses, clamouring to get hold of the $35 single board computer upon its release.

However, a story on the Japanese website XTech Nikkei states that in an email sent to users on 10 August, RS Group's licence to manufacture and retail Raspberry Pi ended effective from June this year. This marks the end of a 10-year licence agreement between RS Group and Raspberry Pi.

Where does this leave RS customers purchasing Raspberry Pi? RS has been taking back orders of Raspberry Pi for some time. The ongoing global supply issues has seen orders slip further and further into the future, many to mid-2023.

In a PC Watch story, it's claimed that there will be no impact on supply for Japan or any other regions. We reached out to Raspberry Pi founder Eben Upton for comment and received the following.

"RS Group has been an important partner for Raspberry Pi, manufacturing our products under licence since launch in February 2012. Our licence agreement with RS Group came to an end in June 2022, with the result that they no longer manufacture Raspberry Pi products. Other licensees, and our own production, are unaffected by this change. We continue to work with our reseller partners to supply Raspberry Pi products to the market, and anticipate no negative impact on availability."



Back when you could get your hands on a Pi...

Ten years of reselling the Raspberry Pi has ended.

Other licensees such as Element14 are continuing to manufacture Raspberry Pi, alongside Raspberry Pi that are made in Wales via Sony.

Supplies of Raspberry Pi boards are still very short, leading to higher prices, purchase limits and anti-scalper requirements to ensure that everyone has an equal chance to grab a slice of Raspberry Pi.

RS Group, which is owned by the OKdo Group, officially commented on the move saying that, "RS Group and OKdo are proud to have been a pivotal part of Raspberry Pi's success and played an active part of the Pi journey for the last 10 years. As Raspberry Pi continues on its growth journey, effective July 1st 2022, we entered a new engagement model, where RS Group is no longer a licensed manufacturer of Raspberry Pi products."

If you were hoping for an easing of supply issues then RS Group had this bad news to add to the cessation of its Raspberry P licence to make and sell the device: "We believe that the chip market will continue to face insufficiency for some time and supply challenges won't unwind materially until the end of 2023.

"We understand that this is a real challenge for customers with the pace of adoption in the smart, (I)IoT, education and maker movement continuing to increase rapidly." LXF

Low-Profile ICE Tower

Les Pounder's Raspberry Pi runs cool. In contrast, Les always feels the heat. Could this new cooler make even our expert chill out this summer?

IN BRIEF

A low-profile, no-nonsense cooler that eschews the RGB aesthetic for straight-up great cooling. The 40mm fan and unusual heatsink arrangement work in our favour and the Raspberry Pi 4 is kept cool, even when overclocked. GPIO access is a stretch, but can be achieved with clever workarounds.

Looking more at home on a muscle car than a Raspberry Pi, 52Pi's latest over-the-top cooler does away with the flashy RGB fan, and gives us a more subtle cooling boost. The Type B Low-Profile ICE Tower Raspberry Pi 4 Cooler (*catchy name – Ed*) looks like an air intake strapped to the SoC of the Raspberry Pi 4.

Assembly is straightforward and you'll soon have your Pi 4 running like Greased Lightning. A thermal pad offers decent contact between the SoC and two 5mm copper pipes that draw the heat away to a heatsink. But where's the fan? On older models the fan was placed on top of the heatsink, but for the Type B we see a 40x10mm fan on the underside of the heatsink. This has the effect of blowing air through the heatsink and keeping our temps low.

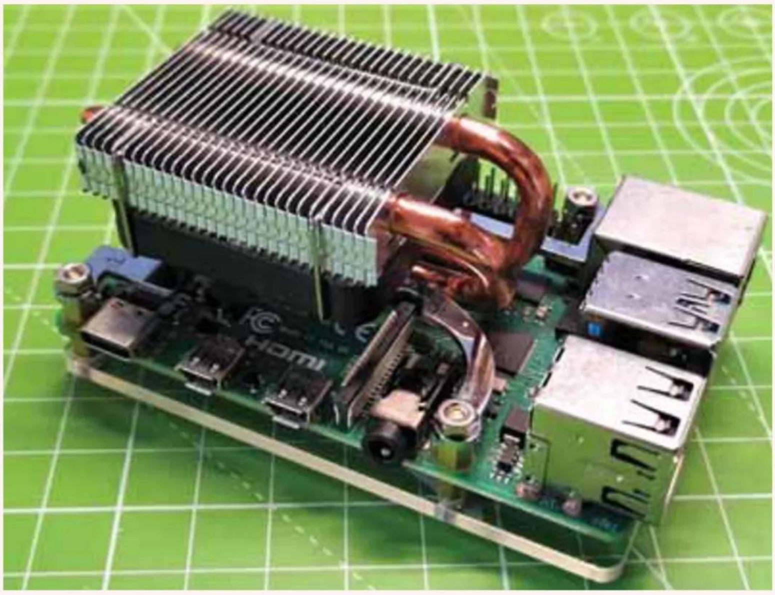
Fan noise is acceptable. You can hear it on your desk, but it's more of a drone than a whine. Trust us, we've tested many Pi cooling kits, the majority of which have a tiny fan that sounds like an angry wasp. Sadly, the fan speed can't be controlled – it's either on or off. Sure, you could rig up your own fan controller, but that means more wires and external control boards.

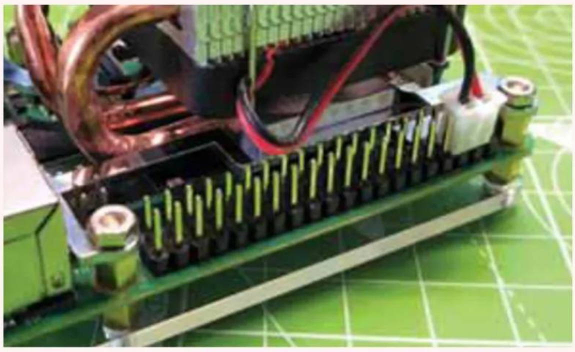
Placed under stress

How good is it at keeping our Raspberry Pi 4 cool? The short answer is very good. We ran a few *Stressberry* tests to push the CPU at the stock speed of 1.8GHz and found that at idle (600Mhz) the CPU was at 30.6°C. When all the CPU cores were under stress, the highest temperature we recorded was 41.3°C.

How does that compare to a Raspberry Pi 4 with no cooling? With the CPU now stock overclocked in recent Raspberry Pi OS release – a decision by Raspberry Pi to get a little more from the SoC – we saw a new idle baseline of 47.7°C. We hit a high of 82.3°C, triggering the CPU to throttle. The top temperature was a little too hot to touch. We also ran the same tests with the fan disconnected, completely passive cooling. The idle temperature sat at 36°C, which was warmer than with a fan, but not too bad. Under load the temperature hit a high of 56°C. How far can we push the Pi 4? We overclocked the Pi 4 to 2.2GHz and ran the same stress test. At idle we saw 32.6°C, while under load it hit a maximum of 47.7°C. So if maximum performance is your goal then this little cooler is for you. The conclusion to these tests is that 52Pi's Type B cooler is great, even when used passively.

Eek out every last bit of horsepower from your Raspberry Pi 4, and keep it cool with 52Pi's low-profile cooler.





GPIO access is possible for jumper wires, but HATs will need a breakout cable and a little rewiring to keep the fan running.

The downside to this cooler is GPIO access. Sure, we could use a breakout to extend the GPIO away from the Pi, but that would become messy because the cooling fan depends on the 5V supply from the GPIO. We can mitigate this by extending the wires, but it won't look pretty. The metal frame that locks the unit to the Pi is designed to fit around the PoE pins, but the unit will block access to all HATs, including the PoE HAT.

We like 52Pi's Type B Low-Profile cooler, because it keeps our Pi cool – even under a heavy load. It also provides plenty of scope for overclocking our Raspberry Pi 4. It has an extreme styling which limits GPIO access, but this cooler is more for servers than makers. LXF

VERDICT

DEVELOPER: 52Pi
WEB: www.52pi.com
PRICE: £19.80

FEATURES	8/10	EASE OF USE	8/10
PERFORMANCE	9/10	VALUE	8/10

It looks like a muscle car air intake, but 52Pi's low-profile cooler keeps your Raspberry Pi 4 cool as you race along.

» Rating 8/10

Pimoroni Trilobot

Les Pounder (quite literally) makes a new friend with a powerful and affordable Pi-powered robot.

SPECS

Chassis: 2x front wheel drive
Ball castor
2x 110:1 DC motors
Grippy wheels
1x HC-SR04 ultrasonic sensor
2x FR-4 PCB
Headers: 40-pin Pi GPIO
2x Qw/St
1x servo
5x garden sockets
1x I2C
1x Pi Camera
Power: via Pi GPIO
Size: 150x106x 53.4mm

Trilobot is a well-designed and easy-to-use robot kit from Pimoroni designed to be Pi 4-powered. For £50 (excluding the Pi, camera and battery), we get a well-designed chassis that uses two FR-4 PCBs to provide a light yet strong chassis containing all of the electronics necessary for building a robot.

The build is straightforward, and is full of exquisite touches. The main PCB is where we see the electronics that make this project. On this board is a single DRV8833 motor controller, the GPIO header for the Raspberry Pi 4, buttons, sockets for sensors and Breakout Garden boards and six RGB LEDs. The Raspberry Pi sits upside down on the GPIO header and provides control and power to the main chassis. Motors connect on the underside of the chassis via custom holders, with two JST connectors on the mainboard providing a simple means to connect the motors.

The ultrasonic sensor, an HC-SR04 variant and official Raspberry Pi camera are held in place at the front of Trilobot using two more FR-4 boards and a few screws. The camera's ribbon cable is neatly routed to the Pi between the top and the main chassis to prevent snags.

One criticism we have about the design is the access to Qw/St connectors. They're behind the ultrasonic sensor and between the wheels, and require nimble fingers to insert the connectors. However, you can take the top layer off for slightly easier access.

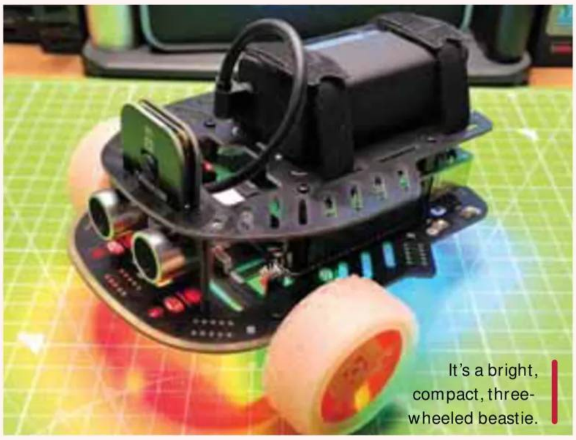
Speaking of connectors, there are five Breakout Garden sockets ready to be soldered. It would've been nice to have a couple of these in the full kit, but it doesn't detract from the overall appeal of the robot. Between the Qw/St connectors is a single socket for a 5V servo. At the rear of the chassis are four push buttons that can be programmed to trigger any action.

Lights, camera, robot!

On the underside are six reverse-mounted RGB LEDs. Reverse mounting means that the LEDs are soldered to the top side of the chassis with a cut-out on the board, enabling them to shine on to the floor. Despite there being only six LEDs they throw plenty of light out.

Powering the Trilobot is a separately purchased USB battery. We received a 5,000mAh USB-C powerbank that provides 5V at 3 amps, which was enough to power our Pi 4 and all the motors, sensors, lights and camera.

You can see where the Pi is attached and powered via the 40-pin GPIO.


It's a bright, compact, three-wheeled beastie.

We wrote a test that saw the motors rapidly change direction while the RGB LEDs were lit at full brightness. From our USB power monitor we saw Trilobot draw 1 amp at 5.2V, giving us 5.2W power draw. So we'd be good for almost an hour's running.

Software is just as refined as the hardware. Parrott and Pimoroni have created a Python 3 package that abstracts control of the motors, RGB LEDs and ultrasonic sensor for ease of use, but retains plenty of configuration options.

The Python software library is mature and it shows that Pimoroni has worked with beta testers from the Raspberry Pi robot community. There's no support for the Raspberry Pi camera in the Trilobot Python library, so we need to use the *PiCamera* or *libcamera* software to stream video or take pictures as our robot runs amok.

The hardware and software are superb, well documented and easy to access. The use of PCBs as building material provides strength and functionality to the robot. The electronics and chassis are one, cutting down on wires and issues. If you're new to robotics, get one of these. If you're a pro, still get one and use it as a stable platform for your next robot project. LXF

VERDICT

DEVELOPER: Pimoroni
WEB: https://shop.pimoroni.com
PRICE: £52.80

FEATURES	9/10	EASE OF USE	8/10
PERFORMANCE	9/10	VALUE	8/10

Trilobot is ideal for beginners and intermediate roboteers alike. It's simple to build and has plenty of expansion options.

» Rating 9/10

Pi PICO W

Stream live data to custom web pages

Les Pounder takes the latest and smallest Raspberry Pi with Wi-Fi and turns it into a £6 web server with live sensor data.

OUR EXPERT

Les Pounder is associate editor at Tom's Hardware and a freelance maker for hire. He blogs about his adventures and projects at bigl.es.

YOU NEED

- **Raspberry Pi Pico W**
- **DHT11 temperature sensor**
- **3 x MtoM jumper wires**
- **Half breadboard**
- **Internet connection**
- **A computer**
- **Get the code from: https://github.com/lesp/LXF-PicoW-DHT11/archive/refs/heads/main.zip**

The £6 Raspberry Pi Pico W has caused quite a stir, offering Wi-Fi for just £2 more. What was usually the domain of the Raspberry Pi Zero W, or other Wi-Fi microcontrollers such as the ESP32 and ESP8266, is now seeing the powerful RP2040-powered board muscle its way in. .

Wi-Fi connectivity provides an easy means to send data, be it to remote devices, or in this case to serve data using a Python-powered HTML server. In this tutorial we'll post live sensor data, taken using a DHT11 temperature sensor, to our own web page that's hosted on a Raspberry Pi Pico W.

Pins matter

The circuit for this build only comprises three wires that connect the DHT11 sensor to the Pico W's GPIO via a breadboard. Looking face-on to the DHT11 (the blue plastic cage) we have VCC (power), Data, No connect, and GND. Place the DHT11 into a breadboard, but make sure that it's not overlapping any pins with the Pico W. Connect GND from the DHT11 to any GND pin on the Pico W. Connect VCC to 3V3, pin 36. Finally, connect data to GP2, which is pin 4. Refer to the circuit diagram in the download file for a high-zresolution image.

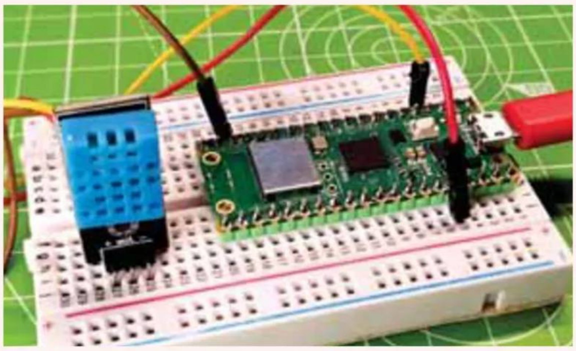
Pico Python

Our Pi Pico W needs the latest version of MicroPython. But we have to use a version specifically created for the Pico W. Visit the official Raspberry Pi Pico MicroPython page (**https://bit.ly/lxf294-micropython**) and download the **UF2** file for Raspberry Pi Pico W.

Holding down the BOOTSEL button on the Pico W, connect the Pico W to your computer using a micro USB lead. Release the BOOTSEL button and a new drive, **RPI-RP2** will appear. Copy the downloaded **UF2** file to this drive and the Pico W will reboot.

Download the latest version of *Thonny* for your OS (**https://thonny.org**) and open *Thonny*. Under Tools, click Options and in the new dialog select Interpreter. Set the interpreter to MicroPython (Raspberry Pi Pico) and set the port to autodetect. If the port isn't autodetected, it can be manually selected. Click OK.

Look to *Thonny's* toolbar and you'll see a red stop sign. Click this to reset the connection to the Pico W. It can also be used to stop any running code.



This simple project is running from a £6 microcontroller that's serving live sensor data to a custom web page.

In a blank file first import four modules of code. These four modules will handle the GPIO, network socket connections, Wi-Fi, and accessing the DHT11 sensor.

```
import machine
import socket
import network
import dht
```

Here we create the HTML for the page. It's saved to an object: `html`. Our first step is to create the HTML head, where we shall set the title of the webpage, and set the page to auto-refresh every 30 seconds.

```
<html>
  <head><title>Raspberry Pi Pico W Temperature Sensor</title><meta http-equiv=Refresh content=30></head>
```

Next we create the body, where the structure for the page is created. We create a div, using Bootstrap's "card" format.

```
  <body>
  <div class="card" style="width: 18rem;“>
```

Inside the `card` div we place an image. Note that this has to be an image from a remote site, not one hosted on the Pico W. Best practice would be to use an image that you have permission to use, and host it either on your own server or GitHub repository. We've also added alt text to explain the image:

```
 <img src="HTTPS LINK TO IMAGE” alt="Temperature Sensor Icon">
```

The next part of the card is its body. Here we set the title to "Live Sensor Data" and then use two paragraphs of text to represent the temperature and humidity. The `{}` will pull data from our sensor readings and drop it into the HTML.

```
 <div class="card-body">
  <h5 class="card-title">Live Sensor Data</h5>
  <p class="card-text">{} degrees Celsius</a>
  <p class="card-text">{} % Humidity</a>
```

We now close the two open divs, and close the body of the HTML document:

```
 </div>
 </div>
  </body>
```

Finally, for the HTML portion of the code, we link to Bootstrap's CDN (Content Delivery Network) and access the latest version of its CSS. This will provide the cool formatting for our page. Lastly, we close the HTML section of code:

```
<link href="https://cdn.jsdelivr.net/npm/
bootstrap@5.2.0/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
gH2yIJqKdNHPEq0n4Mqa/
HGKIhSkIHeL5AyhkYV8i59U5AR6csBvApHHNl/
vI1Bx" crossorigin="anonymous">
</html>
“ ” ”
```

Next we create an object, `sensor`, to connect our code to the DHT11 sensor connected on GP2.

```
sensor =dht.DHT11(machine.Pin(2))
```

To connect to Wi-Fi we first create an object, `wlan`, to control the Wi-Fi chip. We then set this chip to active:

```
wlan =network.WLAN(network.STA_IF)
wlan.active(True)
```

Using our Wi-Fi access point SSID and password we then connect:

```
wlan.connect("YOUR SSID",“PASSWORD")
```

We now move into the section where we set up our sockets, which will listen for any incoming connection:

```
addr =socket.getaddrinfo('0.0.0.0’,80)[0][-1]
s =socket.socket()
s.bind(addr)
s.listen(1)
```

Inside of a loop we set the socket to accept a connection, then create a file that will be our HTML page:

```
while True:
   cl,addr =s.accept()
   cl_file =cl.makefile('rwb’,0)
```

We read the lines that make up the HTML page. If there are any line breaks our carriage returns, then the loop will break:

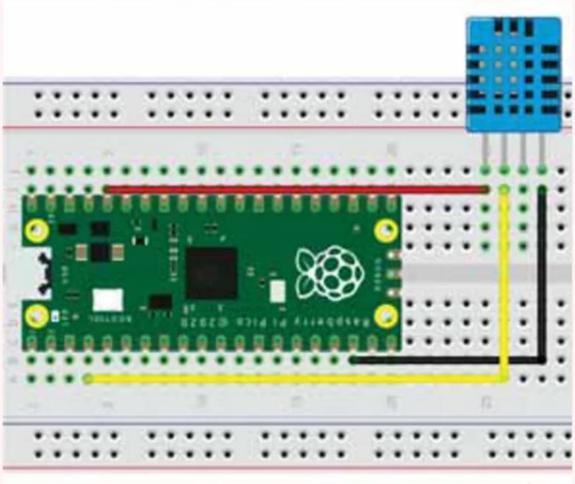
```
   while True:
      line =cl_file.readline()
      if not line or line ==b'\r\n':
         break
```

Before we can get sensor data from the DHT11 we need to measure the temperature and humidity. If we skip this line, our sensor won't work.

```
      sensor.measure()
```

Create two variables, `temp` and `humid`, and in each store the current temperature and humidity detected by the DHT11:

```
      temp =sensor.temperature()
      humid =sensor.humidity()
```



The DHT11 needs only three connections: VCC for 3V voltage, GND, and a data connection that sends temperature and humidity data to the Pico W.

We then create an object, response, and use that object to drop the temperature and humidity data into the HTML. These values are placed in the `{}` that we created in the `card-text` classes inside our HTML code:

```
   response =html.format(str(temp),str(humid))
```

Finally we ensure that our HTML page is compliant with HTTP content types and close the connection. The connection will automatically reestablish itself in 30 seconds, as per the page refresh:

```
   cl.send('HTTP/1.0 200 OK\r\nContent-type:text/
html\r\n\r\n')
   cl.send(reponse)
   cl.close()
```

Save the code as **web_temp.py** to your Raspberry Pi Pico W. Click Run to start the code. Find the IP address of your Pico W using your router admin page, or an application such as *Fing* for Android.

In a browser on another device, go to the IP address and after a few moments you'll see live sensor data, refreshed every 30 seconds. LXF

QUICK TIP

Sometimes you may need to reset your Pico W, but there isn't a reset button. Just use an extra male to male jumper wire to quickly touch GND and RUN (pin 30) to reset the Pico W.

» WHICH SENSOR IS RIGHT FOR YOU?

The ideal temperature sensor for your project depends on what level of accuracy you require. For hobbyists, the DHT11 is more than enough. It can detect between 0 and 50°C temperature readings ±2°C accuracy, and 20-80 per cent humidity readings with five per cent accuracy. We can only take a reading every one second, though.

The DHT22 looks similar to the DHT11, but it can works between -40 and 80°C with an accuracy of ±0.5°C accuracy. It can take 0-100 per cent humidity readings, with two to five per cent accuracy. So for more scientific projects, the DHT22 is the better option.

An even more precise option is an AHT20. This I2C-based device has MicroPython libraries that can be used to simplify working with it. Accuracy is ±0.3°C between -40 and 85°C. Relative humidity range is 0 to 100 per cent with ± two per cent accuracy. The AHT20 can be found on Adafruit and Pimoroni's website for around £5.

If you need a waterproof temperature sensor then consider the DS18B20. It can measure between -55 and 125°C, with 0.5°C accuracy between -10 and +85°C. Obviously there's no humidity sensor, but for temperature measurements then this is a great sensor. It can be used to monitor aquariums or pools or even your bath water.

» **MAKE US ALL HOT & BOTHERED** Subscribe now at **http://bit.ly/LinuxFormat**

KODI (MATRIX)

Credit: https:// kodi.tv

Create a Pi-based media centre with Kodi

Kodi is an open source media centre. **Michael Reed** installs it on his trusty Raspberry Pi for some couch-based entertainment.

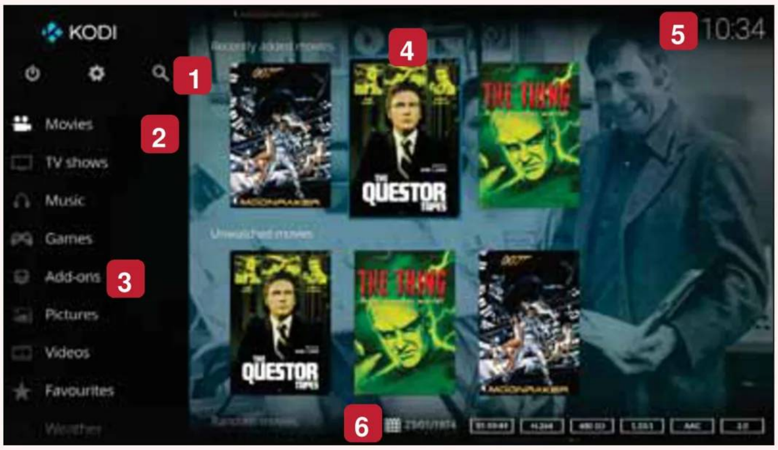
OUR EXPERT

Michael Reed has been spending so much time with Kodi of late that his mum thought he'd finally got a girlfriend.

Kodi is a cross-platform application that can turn a device such as a Raspberry Pi into a fully fledged media centre. It presents the user with a simple interface to move between different types of media that are automatically categorised. It can handle video and audio files that are located on multiple sources, and it can even access streaming media sources that are located on the internet. Add a remote control and you'll be able to launch movies, TV shows and music, play DVDs and access shared drives. You may never need to leave the couch again!

We're going to take the approach of installing *Kodi* next to a desktop Pi OS (Raspberry Pi OS) so that you've got the best of both worlds: a full desktop computer with up-to-date software and a media centre in one machine. These instructions should be easily adaptable to any Linux-based Raspberry Pi OS. What you get, compared to a prebuilt, smart TV-style device, is the chance to customise the services on offer and add any niche services that you may need.

EXPLORE THE KODI INTERFACE



1 Key icons
Power, settings and search icons. Navigate to the top to get to these.

2 Heart of Kodi
The main sections. Move up and down to navigate these. There's also some room for customisation here.

3 Customise the program
The Add-ons icon takes you to a sub-area where you can browse to specific user interfaces for each add-on, such as streaming services.

4 Media library
In each section, a curated media library is presented. Click on the items or hover over and bring up the context menu.

5 What's your status?
This status area shows things like the current time. When you're watching a movie, it even tells you what time the movie will finish..

6 Extra information
When you browse to media, this area gives you information such as the total media, the codec, aspect ratio and the resolution.

A Kodi to call my own

First things first, we need to install the system. *Kodi* is a lightweight piece of software, but we'd recommend one of the more powerful models of Raspberry Pi when building a media centre. This ensures smooth playback along with greater speed when adding material and carrying out utility operations. For that reason, we'd recommend a Raspberry Pi 3 as a minimum for this project, and with a Raspberry Pi 4 or 400 you'll have even more fun.

These instructions assume that you have the latest version of Raspberry Pi OS (formally Raspbian) installed and ready. However, most of what we're describing can be adapted to other Linux-based Pi operating systems.

If you're installing from fresh, follow the usual procedure to install Raspberry Pi OS. Download the latest OS image (**www.raspberrypi.com/software**), and then write the image to an SD card using your tool of choice such as the official *Raspberry Pi Imager* application that's in the repository of most Linux distros.

You can locate the actual media that you want to access onboard the SD card inside your Pi, an attached external drive or on a network. You can even use a mixture of all three types of storage.

Get mixing and matching

Part of the beauty of setting up your own media centre is that you can design your own mix of components. A common setup is to plug the Pi into your TV using a spare HDMI slot. All Raspberry Pi computers (apart from the Pi 400) are supplied in the form of a single board. For that reason, if you're making a media centre then you'll probably want to purchase a case for your Pi. If you're going to be running a media centre 24/7, you might prefer a case that includes a heatsink, giving extra peace of mind from passive (silent) cooling. You don't want everything to crash because of overheating when you're halfway through watching a film.

You might be building a Pi media centre that works without a keyboard and mouse, as part of the eventual plan, but we'd recommend that you attach both while doing the initial setup. In addition, we're assuming that you have a Linux desktop computer attached to the same network as a Pi.

Use SSH

For work that has to be carried out at the command line, you might consider logging into your Pi via *SSH* from another computer. This is easily set up on Raspberry Pi OS (Launcher Menu>Raspberry Pi Configuration and then Interfaces tab and enable SSH). Log in from a computer on your network with `ssh <Raspberry Pi username>@<Raspberry Pi IP address>` and then enter your password. Using this, you can access the command line of your Raspberry Pi without having to exit *Kodi* to do so.

This isn't the most secure of setups, so either set up a secure SSH passworld-less login or simply disable SSH using the user interface when you're not carrying out maintenance work. In any case, no one can access your Pi from outside of your network.

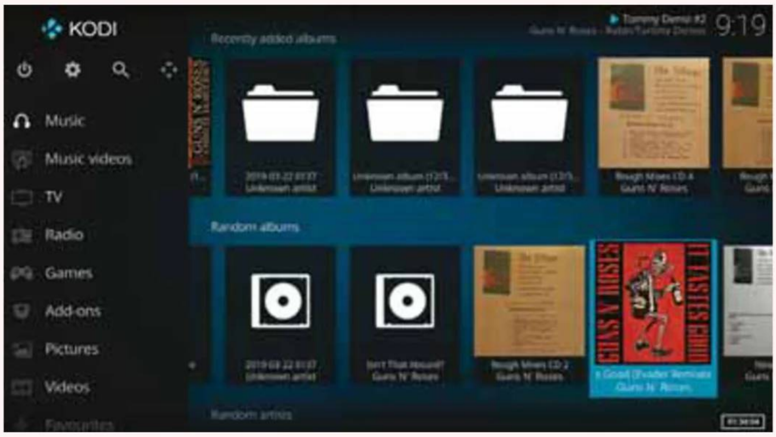
Installing Kodi

Installing *Kodi* on Raspberry Pi OS is simple. Type `sudo apt install kodi` to install the latest version of *Kodi* that's present in the repositories.

Once you've installed *Kodi*, it should be launchable like any other application (launcher>Sound & Video> Kodi). You can even launch it from the command line (local or *SSH*) by typing `kodi &`. Launch it to make sure that it's working.

You'll probably be able to get a rough idea of how *Kodi* works at a first glance. That is, after all, a large part of the appeal of a kiosk-style media centre. Along the left-hand side, we have a set of sections to which you can add media and configure separately. You can also add extra sections and remove them as needed.

You can navigate the interface using the cursor keys on the keyboard attached to the Pi or even by using a mouse. However, hopefully, you've got something better to use such as a dedicated remote (*see box, page 49*) or an attached game controller.



Use your TV as a jukebox with help from Kodi. Makes recompiling the kernel fly by.

Adding media

It's time to start adding content to *Kodi*. You might already have a video collection consisting of films and TV shows. If not, you can rip them from your DVD collection using a tool such as *Handbrake* (**https://handbrake.fr**). *Kodi* has some rules about how videos should be named. If you stray from this naming convention, you can still watch the video files in *Kodi*, but you can't add them to its library.

In the case of a film, the name of a film should be followed by the year of the film in brackets. For example, **Moonfork (1979).mp4**. TV shows use the format of: [name of show] [(year of show)] [series] [episode]. For example, **LXF Towers (2005) S1E1.mp4**. Kodi then uses a series of 'scrapers' – scripts that consult with external databases – to attempt to find out exactly what the film or TV show is. *Kodi* can cope with nested directories, in case you're the organised type. Create separate directories for films and TV shows and copy your video files into them. The same goes for music files in MP3, WAV or FLAC format. See further on in this article if you want to add directories that are on your network.

Add the directories that you have created as 'sources'. It's easy to mix and match between internal, external and network resources, adding multiple media sources at once. From the home page, go to Settings> Media>Add videos... This brings up a dialog. From here, click Browse and navigate to the directory containing video files. Note that when you find the correct directory, the main part of the dialog doesn't show any

QUICK TIP

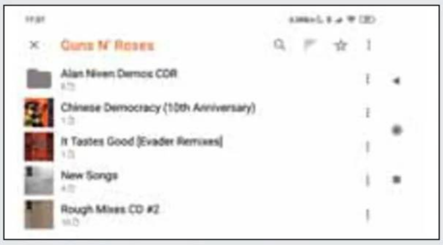
Kodi uses a major and minor version number. Each major version also has a code name. At time of writing 19.x (Matrix) releases are the current stable releases, while 20.x (Nexus) is the development branch of Kodi.

» GET UP YOUR OWN MEDIA SERVER

As well as being a media centre, *Kodi* can work as a media server. It does this by using a standard called DLNA that's widely supported by devices such as tablets and game consoles.

Enable DLNA by visiting Settings> Services and going down to the UPnP/ DLNA tab. UPnP is the service by which devices can discover an active DLNA server, so enable that as well. In addition, enable the 'Share my libraries' option to share all of the content that your Kodi setup can see.

Once they've been shared, how you browse these resources depends on the client device. For example, if you used the desktop version of *VLC*, the folders and media files should be locatable by showing the playlist (View>Playlist) and then selecting Universal Plug'n'Play on the side bar. Note that the interface for browsing content served over DLNA tends to be bare-boned, but it should work for all audio and video files that the client device would be able to play. We had success using the desktop and Android versions of *VLC* and a PlayStation 3 games console with *Kodi* as the server.


Guns N' Roses
Alan Niven Demos CDR
Chinese Democracy (10th Anniversary)
It Tastes Good (Evader Remixes)
New Songs
Rough Mixes CD #2

Browsing the media that Kodi can access via VLC on an Android phone.

» GET MORE "ENTERTAINMENT"! Subscribe now at http://bit.ly/LinuxFormat

QUICK TIP

What about the games? At time of writing, the support for retro gaming emulators isn't quite ready for operating systems based on Debian Bullseye. So, we decided not to bother with games this time around in favour of an up-to-date desktop OS.

files. Click OK. The next dialog enables you to set the content type. Set this to Movies, TV Shows, Music and so on as appropriate. This content should now be navigable via the home page of the system.

Adding network resources

Let's take the example of a shared media directory called **MyVideos** that's located on a NAS (Network Attached Storage) drive and shared via the Samba protocol. *Kodi* does have some network discovery features, but we've found that the most robust technique for adding network shares is to add them to mount points, as though they were standard directories local to the Raspberry Pi.

Find out the IP address of your NAS. This is the address that you use to navigate to the NAS web interface. Either use your main computer, the Pi or the web interface of your NAS to create a directory called **MyVideos** on your NAS, and put at least one file into it for testing.

On your Pi, create a directory in your home directory that's also called **MyVideos**. This is the mount point.

Type `sudo mount -t cifs //<IP address of NAS>/MyVideos. /home/<your Raspberry Pi username>/MyVideos`. If the directory now contains the contents of your remote directory on the NAS, things are working properly. Make this connection permanent. Type `sudo nano /etc/fstab` to edit that file and paste in the following line:

```
//<IP address of NAS>/<name of folder> <full path of mount point on Pi> guest,uid=1000,gid=1000,iocharset=utf8 0 0
```

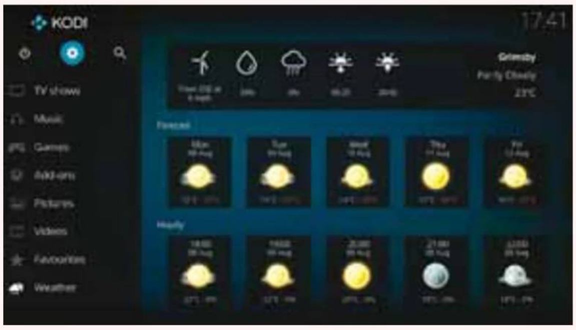
Type `sudo mount -a` to test out this configuration is accepted and works.

There's one more thing you have to do to make this permanent on Raspberry Pi OS. In the Raspberry Pi configuration tool (Launch menu>Preferences> Raspberry Pi Configuration), enable Network at Boot. Try rebooting to make sure that the configuration sticks. Using this technique, you can add the root directory of your NAS that contains your media, or you can add multiple directories with different types of media.

Log into the web interface

The *Kodi* web interface enables you to control *Kodi*, see the current status and carry out administration duties from another device. You should be able to navigate to most areas of the system from here, and the user interface roughly mirrors that of the actual *Kodi* user interface. Occasionally, it's helpful to be able to cut and paste text (such as passwords and URLs) from your computer and send it to *Kodi*, and the most convenient way of doing this is via the web interface.

Within the *Kodi* interface, navigate to Settings> Services>Control and select 'Allow remote control via HTTP' to enable the web interface. Note the port number, and it's a good idea to choose a password and a username at this point. You can now navigate to the user interface by putting **<IP address of Raspberry Pi>:8080** into the URL of a browser.



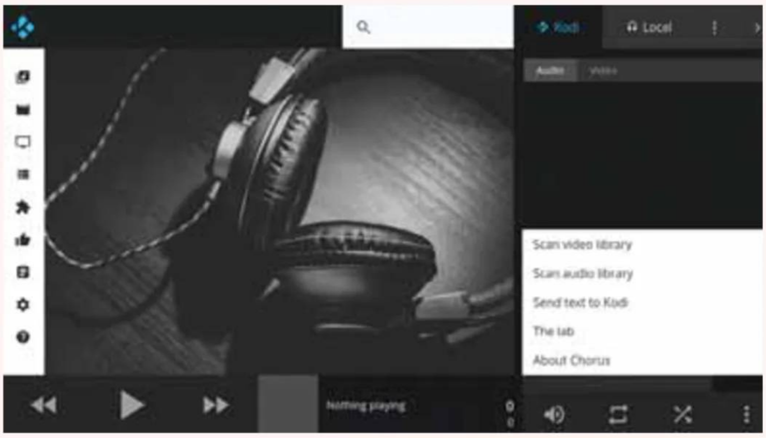
The weather applet is easy to set up (Settings>Services>Weather). We found that the 'multi-weather' server worked best for UK weather.

Add streaming video support

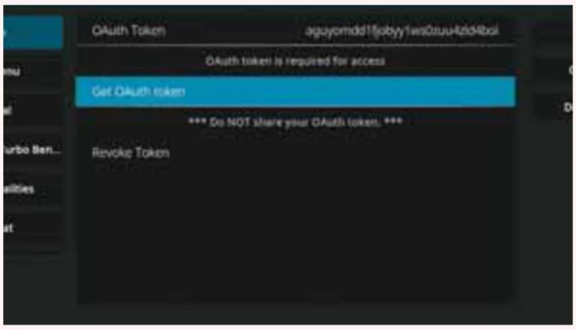
By and large, we recommend a 'suck it and see' approach to add-ons within *Kodi*. Many of the plug-ins are unofficial and therefore unsupported by a given service, and the developers have to play catch up when the service itself changes. To add a streaming service, look up the *Kodi* add-on for it (Settings>Add-ons> Search).

Adding support for the Twitch video streaming service is a good example of the procedure to add streaming media sources to *Kodi*, and it's a good example of *Kodi's* add-on system. Add the Twitch add-on to *Kodi*. Do this by going to the settings icon on the home screen and navigating to Add-ons>Install from repository>Video add-ons>Twitch. On the Twitch plugin page, click install. This should put you back on the Video Add-ons page.

If you'd like to log into your normal Twitch account to have access to your subscribed feeds, click Twitch again, and this time, click Configure on the Twitch plugin page. On your desktop computer, log into your account



The Kodi web interface shows the current status of Kodi and offers some simple controls and utilities, such as text cut and paste.



OAuth token is required for access

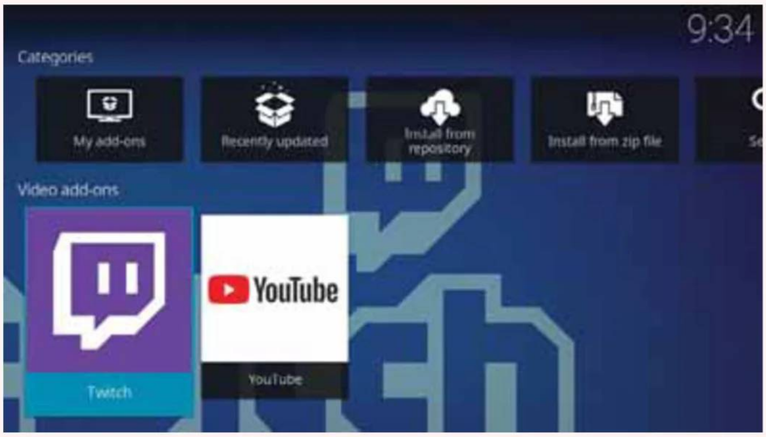
*** Do NOT share your OAuth token. ***

Configuring an add on. Most of the settings can be navigated with a remote, but it helps if you have a QWERTY keyboard from time to time.

in the normal way. Back in *Kodi*, on the Twitch configuration page, select the Get OAuth token option. This will give you a link that you then type into a web browser on your desktop computer. Ultimately, this 'token' needs to be pasted into a field within *Kodi*. Because *Kodi* doesn't have great support for this feature, this is where the web interface comes in. Simply navigate to the OAuth Token field and paste from your desktop computer using the 'Send text to Kodi' option that's located within the three-dot menu at the bottom of the web interface.

Now it's time to test the *Kodi* plugin. Go back to the home screen and scroll down the side icons until you get to the Add-ins entry. In the main area, you should be able to navigate to the Twitch add-on. Click this and you'll be presented with a set of 'directories' that represent categories that represent live channels and games and individuals that you're currently following from your account. We found that performance on a site like Twitch was good from within *Kodi*. By comparison, the actual website has a rather busy layout and runs a little slowly when running a web browser on the Pi.

Adding all streaming services follows a similar procedure, and as you can see from the Add-ons section, a huge number of sites are supported including YouTube and Netflix.



Online streaming services such as YouTube and Twitch are mostly added via Kodi's add-ons.

CD and DVD support

If you've got traditional digital media lying around, you might consider adding an external DVD drive to the setup. However, because a device like that has moving parts, it tends to need a lot of power and will probably require a powered hub, adding to the clutter around the unit. For example, we found that a Raspberry Pi 400 with the official power supply kept disconnecting itself when playing DVDs until we added a powered hub. However, once you've got it set up, it couldn't be much simpler to use. When you insert a disk into the drive a 'Disc' icon pops up amongst the other ones. Click that, and you'll have a fairly standard DVD player experience. However, for convenience, you might consider ripping physical media directly into the video source directories that *Kodi* is connected to, using an application on a desktop computer.

Moonraker (1979)
007
MOONRAKER

Browsing some automatically scraped information about a film (select Movie and press the context menu button).

Finishing up

You might like the system to start up with *Kodi* rather than the underlying Raspberry Pi OS desktop, and you do this by editing a file with the *Nano* text editor. Type `sudo nano /etc/xdg/lxsession/LXDE-pi/autostart` and add the line: `@kodi` at the bottom. Press Ctrl+o to save the file and Ctrl+x to exit *Nano*. That way, when you exit *Kodi* (Power>Exit), you'll be returned to the Raspberry Pi OS desktop. When you shutdown (Power>Power off system), the entire machine will be shut down.

Back on the Raspberry Pi OS desktop, you should be able to access most of the services and applications of a normal desktop computer. As we discovered back in **LXF288** in our feature on using a Raspberry Pi as a daily driver machine, a top-end Pi is quite formidable in this role. **LXF**

QUICK TIP

You can discover the IP address of your NAS by using Nmap. Install it with `sudo apt-get install nmap`. Type `sudo nmap -sn [your network range]` to list the devices that are connected to your local network.

» MAKE USE OF A REMOTE CONTROL

You can navigate *Kodi* by using the cursor keys on a QWERTY keyboard if you have to, but for media viewing you might consider buying a dedicated remote. We were able to find units that only cost about £10 and even included a miniature QWERTY keyboard on the back.

To test for compatibility, the simplest method is to search the reviews on sites such as Amazon and eBay and look for mentions of other people using the remote with *Kodi* and with a Raspberry Pi. Again, the cheap unit we bought was immediately detected by *Kodi* on launch without us having to do anything else. Even if the remote hadn't been detected it acted like a keyboard sending out standard keypresses, so could still have been used.

It even included 'air mouse' facilities, which were awkward in comparison with a mouse, but usable. You might find that a unit like this is everything you need to use your media centre, even if you intend to use it for some light desktop computing in normal Raspberry Pi OS.

A game controller can also be used as a remote. Note: we had to add an extra package with `sudo apt-get install kodi-peripheral-joystick` to get it running.

Controller Configuration

Selecting from the array of accepted controllers.

Living room
Temperature
24°C
75°F
Front door
Kettle
Oven
Fan
Cooling
Schedule
Done

SAVING THE SMART HOME

The smart home is dying but **Matt Holder** is here to save the day by showing us how open source can keep the smart-lights on…

This month we're going to look at some of the reasons smart homes can work against us. We'll be considering hardware issues running self-hosted systems as well as what happens if vendors decide they don't want to provide certain services any longer, if they want to introduce a subscription charge or that they simply become bankrupt and can no longer function. Finally, we'll be looking at ways that we can mitigate these problems by having stable equipment at home and running services to keep traffic in our home networks.

The first thing to discuss is unreliable hardware running any self-hosted systems. Your author experienced problems for a number of months when running *Home Assistant* on a Raspberry Pi with micro-sd cards, followed by reliability problems using a virtual machine on an old desktop PC. Understandably, during this time the spousal disapproval factor was a thing, while we were both very frustrated with constantly needed reboots, failing micro-sd cards and so on. Lesson one is to purchase suitable kit to ensure a good experience. Your author's home runs a Raspberry Pi 4 8GB as the main host and is booted from a 240GB USB SSD, which has proven to be a rock-solid combination and provides enough RAM and processing power for everything that's required.

One of the earliest protocols to be used for home automation purposes was X10. This was first designed in the mid-1970s and could operate both using radio frequencies to communicate with devices and also as a signal injected over the mains power lines. This was originally controlled using dedicated controllers, but devices with serial port connections eventually became available, which were used to enable devices to be computer controlled. While being largely superseded by other protocols, X10 devices are still used around the world and can be operated with large open source home automation systems, which will be discussed later on in this article.

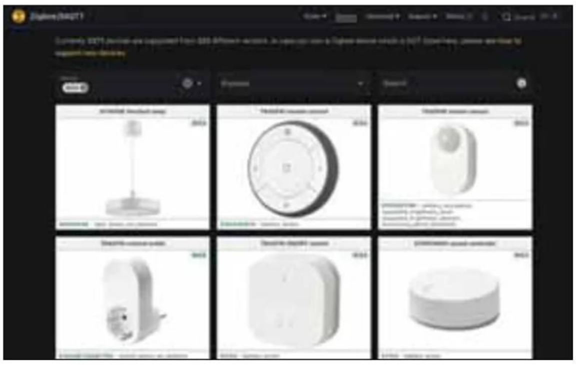
The past number of years have seen companies fall into the habit of everything being cloud-controlled when it really isn't necessary. Local and open APIs are the best way to communicate with your smart home kit because it removes the risk that the supplier will make a change in a firmware update and you're then locked out.

Rising costs

Running a cloud service for a large number of customers is expensive and possibly not properly considered when deciding to release a device that's dependent on the cloud. The short-term boost to a company from selling a device for a profit is quickly eaten up by the costs of running databases, servers, load balancers and other cloud-based services or servers. Over the years there have been a number of announcements from companies that they are either shutting down or beginning to change the previously free to a charged-for cloud service. This is a real insult to customers who were promised a service and the reason that local APIs and self-hosted control software offer a level of control against this happening to you.

During 2020, Wink, which offers cameras, sensors, lighting, smart plugs, locks and thermostats, made changes to its offering. This meant that a paid subscription was needed to take advantage of previously free features. Items that were lost were the ability to have integration with popular voice assistants, third-party integrations, groups robots, the ability to add devices, perform firmware updates and more.

Belkin's Wemo brand of cameras were quickly discontinued. Shortly after this, in 2020 it was revealed that cloud accounts and internet access would be required for Belkin's other devices to function.



Check out if your devices are compatible and open with the Zigbee2MQTT system, such as these IKEA products.

Perhaps the highest-profile event to hit the home automation community was when the US company Insteon shut down. Insteon provides outlets, switches, keypads, sensors, thermostats and remotes and utilises, in a similar way to X10, communication via RF and injected over the powerlines. At the time the company closed, devices were no longer controllable via the official app and a lot of users were bound to have been angry and felt helpless by what had happened. At the time of writing, hubs are back online and users are still able to use their devices. This is because the company has been resurrected by a number of users who have bought the company. Just shy of $40 per year is required to purchase a subscription to help the company return to a viable status. At the time the company was declared bankrupt, the HomeAssistant developers released an Insteon integration to enable users to continue using their equipment that could be controlled via a USB adapter.

CREDIT: Ecobee

The Ecobee thermostat is the hottest thing in the open smart home world.

TP-Link produces a range of Kasa smart plugs. TP-Link addressed "security vulnerabilities on the plug for the local management... TP-LINK UK (@TPLINKUK) November 17, 2020" by releasing an update that locked users out of the local APIs, meaning that the official app was the only way to go. Thankfully, TP-Link reversed the decision and released another update, which added the functionality back in again.

As most of us are aware, the website **https://killedbygoogle.com** lists products that have been introduced by Google and then subsequently cancelled. Following the purchase of Nest, Google made changes to the process of obtaining API access. This was a first warning sign of the usage of Nest products within an open smart home environment.

Finally, it has recently been announced that Hive will be removing support for some of its cloud-connected IoT devices between 2023 and 2025. In 2023 the leak detector sensors will no longer function as expected and in 2025 the cameras being discontinued will, essentially, become bricks.

ZigBee and Z-Wave

A lot of devices being sold at the moment utilise the ZigBee standard. This is a protocol specifically designed for home automation. It uses the same frequency as

QUICK TIP

Smart plugs, pre-flashed with Tasmota can be purchased from UK (www.mylocalbytes.com) and US (https://cloudfree.shop) suppliers. Other items are also available. Tasmota details can be seen at https://tasmota.github.io/docs.

VPN ACCESS TO CONNECT REMOTELY

The main selling point of the cloud-connected smart home is that devices can be controlled wherever we are. Nest, for example, is one such service, providing cameras, thermostats and other devices. Lots of services that offer this functionality aren't strictly required and it would be much better to keep traffic within the home network and connect using a VPN when it's needed. Many different options are available including *ZeroTier*, *WireGuard* and *Tailscale*. *Tailscale* can be installed on pretty much any device and operating system, and the client for mobile devices works very well.

Tailscale works by creating a mesh network between devices that can be connected via IP address or DNS names. Once the connection route has been established, *WireGuard* is used to secure traffic between the nodes. The *Tailscale* team offers a very generous free tier, enabling up to 20 devices to be registered. An added bonus of using this VPN is that you can route web traffic from your phone through the VPN and it'll appear to be coming from your home's IP address. Other useful features include ACLs, to enable a friend's access to one of your *Tailscale* nodes and the ability to use MFA, by allowing sign in with a Google (or similar) account.

»

QUICK TIP

Volumio can be used to use a Raspberry Pi as a streaming audio player – see https://volumio.com/en for details.

2.4GHz Wi-Fi and devices build a mesh network that all communicate with each other. Mains-powered devices, such as bulbs or smart plugs, can act as routers that enables the network to have an extended range and function with more stability and reliability. Battery-powered devices last for months before new batteries are needed, due to Zigbee's low-power consumption.

Lots of devices will also either come with a hub or state that a manufacturer-specific hub is required. This isn't necessarily true and using a USB ZigBee controller and a project, such as ZigBee2MQTT, can remove the reliance on proprietary applications and hubs. Once configured, ZigBee devices can be paired with your controller and operated in this manner. Examples of ZigBee controllers include the ConBee II stick and the Sonoff CC2531. The latter device will cost less than £10, but note that performing firmware updates is tricky without buying extra equipment.

Smart home equipment that uses the ZigBee standard includes the Sonoff range of switches, bulbs and sensors, the Hue range of lighting equipment and the IKEA Tradfri range, which includes switches, bulbs, smart plugs and motorised blinds. Some Smart Energy Meters operate using the ZigBee protocol, but changing the settings of these isn't recommended due to the possibility of being accused of tampering with and trying to change your meter readings!

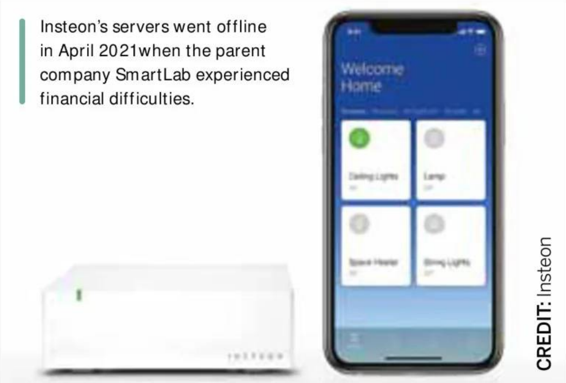
Belkin pulled support for its NeoCam service.

CREDIT: Belkin

Ride the wave

Z-Wave is another protocol, which is similar to ZigBee. Z-Wave, however, uses the 800-900MHz frequency, so can sometimes be better in terms of interference between smart home devices and Wi-Fi networks. Projects such as Z-WaveJS can be used in a similar way to Zigbee2MQTT. An example of a Z-Wave USB stick is the Aeotec Z-Stick 7. Equipment using the Z-Wave standard includes bulbs, smart locks, switches and many other devices.

Tasmota is an open-source project that can be used to provide alternative firmware for devices that utilise the ESP range of microcontrollers at their core. When installed and configured, devices can be controlled using the built-in web interface and MQTT. The *tuya-convert* project can be used to upgrade the firmware on a large number of tuya-made devices without needing to open them up. Tuya is a large manufacturer of unbranded smart home gear that many manufacturers purchase, brand and resell. When using *tuya-convert*, the software acts as a Wi-Fi access point, which the tuya devices automatically connect to. This mimics how firmware updates are carried out when tuya devices are being prepared to be shipped. *Home Assistant* does have an official integration for Tuya devices that can work locally and via the cloud service.



Insteon's servers went offline in April 2021 when the parent company SmartLab experienced financial difficulties.

CREDIT: Insteon

MQTT is a messaging system that provides the glue to stick together multiple different systems. When configured, the MQTT broker enables devices to publish messages using an addressing system similar to a URL, and devices can then subscribe to these messages. Using this system a smart plug can publish its status and subscribe to another topic, where it can then act on messages it receives from another piece of software, such as a command to switch the socket on.

Start a project

Projects exist to enable smart home fanatics to be able to bring many different manufacturers, protocols and devices together and use data from them all to control how our smart homes can work to help us.

Home Assistant has an enormous number of add-ons, integrations and community members. It's written in Python, was first released in 2013 and can be installed using its provided operating system, using containers or straight on to a base OS. *OpenHAB* is a Java project that goes back to 2010 and has a similar architecture in terms of add-ons and having its own operating system available. *Domoticz* is the third project that can be used to control your smart home and versions exist for Windows, Linux and for the Raspberry Pi's Arm version of Linux. *Domoticz* can also be run inside of a container. All three projects have an active community ensuring that new users can increase their knowledge and progress with their usage of the software.

» WHY DOES ANY OF THIS MATTER?

Matter is a new standard that promises to improve interoperability between vendors of smart devices. Support for Matter is high across the industry and the following organisations have all joined a group to discuss how to move forwards: Amazon, Apple, Samsung, Google, IKEA and the ZigBee Alliance. Matter will build on the ZigBee specifications. When Matter is being used by a large number of suppliers it means that there's less to worry about interoperability between different manufacturers and the need for hundreds of apps to control them all.

Because they all speak the same language, controlling them all from a central solution should be a great deal simpler. The release of the Matter protocol, along with the finalised SDK has been delayed until the autumn of 2022, due to unprecedented interest, with devices hopefully available shortly afterwards. The Home Assistant team held a stream (*see tip right*) where users could follow along with a microcontroller board and create a basic hardware device, utilising the Matter protocol. The future looks good with so many large vendors working in this new specification.

In recent years the market has been flooded by vendors offering easy-to-connect CCTV cameras. These generally utilise battery-powered cameras that connect via Wi-Fi directly to their cloud service. A lot of vendors charge for the device and offer a basic tier of their subscription service for free. Paid-for upgrades are generally available, which allows for footage to be retained for a longer period of time. These devices are convenient in the short term because they're quick to mount and configure, but less convenient in the long term as they'll need charging and reconnecting to the Wi-Fi if your router is changed. By selecting a camera that outputs an RTSP stream, it can be connected to a number of open source CCTV NVRs that can be self-hosted and offer much longer retention periods for footage. Wired and wireless cameras exist and this author's perspective is that it makes more sense to put in the effort to set up at the start of the process, rather than needing to charge batteries. Examples of CCTV NVR solutions include *ZoneMinder*, *MotionEye*, *Shinobi* and *Frigate*. See last issue to learn more about *Frigate*.

A large number of manufacturers have released cloud-connected thermostats. Google's Nest, Hive from British Gas and Honeywell offer cloud-connected and local thermostats. Ecobee thermostats don't require cloud connection and another option is to create your own using a microcontroller board, temperature sensor, ESPHome and integration with *Home Assistant*.

Earlier on in the article we discussed being able to flash the Tasmota (ESPHome is also possible) firmware on to Tuya devices using *Tuya-convert*. Dependent on firmware versions installed it's not always possible to use this tool. It's also tricky to purchase new devices knowing whether they can be reflashed or not. A better option would be to buy smart plugs that are pre-flashed with the Tasmota firmware. See Quick Tip (*page 51*) for details on where to buy such plugs.

Speaking of which…

Voice assistants are another area of huge interest, due to how useful they are. We must always keep in mind whether information is being mined from the questions we ask, given who makes the smart speakers. For example, Google's main revenue stream is in advertising and knowing what to advertise to us is in its best interests. Projects such as MyCroft and Almond exist to offer this functionality away from large companies and their interests in us.

Smart speakers are also hugely popular devices in our homes and a large range of manufacturers and standards are available. For example, Sonos is very popular and offers great-sounding devices. A lot of voice assistants can also be used as speakers, enabling music to be played from multiple different sources. Some of these sources, such as Google Cast and Sonos, can be operated from our home automation systems.

amazon alexa
tp-link
Smart Wi-Fi Plug
Works with IFTTT

CREDIT: TP Link

TP Link made security changes that locked users out of the Kasa API.

Wink switched from a free to fully paid-for subscription service, cutting off anyone who didn't sign up.

The DLNA alliance was formed in 2003 to create an interoperability standard for playing music from different sources to many different speakers. This standard is still used today and integrations exist for the major desktop and mobile operating systems as well as home automation tools. While on the subject of music, recently the Music Assistant project has launched, which is an add-on for Home Assistant, which enables music from multiple sources, *Spotify*, local files and internet radio stations to be added to playlists and streamed to players across the home. It's been announced that there's been integrations with the ESPHome project, where microcontroller boards can have audio DACs connected and be connected to passive speakers, so that they can be made into smart devices. Also available, in a similar way, are boards to enable Raspberry Pis to act as audio players.

THE DANGER OF SAYING TOO MUCH

> "We must always keep in mind whether information is being mined from the questions we ask, given who makes the smart speakers."

Using a system such as OpenHab enables many systems to work together to allow automations and notifications using multiple sources. For example, notifications could be sent based on the pressing of a ZigBee doorbell button, which also includes a snapshot from a CCTV camera. Your imagination is quite literally the limit of what can be achieved.

This article has introduced some considerations that will help you to ensure that your journey with smart home equipment is as long-lived and interoperable as possible, while ensuring that as much data as possible is kept within the boundary of your home network. There are many communities available should you want to learn more about any of the technologies discussed in this article. As ever, enjoy exploring your smart home. LXF

QUICK TIP

The Home Assistant team held a demo where users could follow along with a microcontroller board and experimental Matter support – see www.youtube.com/watch?v=9fOHBI5w0_k

KMON

Credit: https://github.com/orhun/kmon

Manage kernel modules

There's a lot to be said about finding inner peace through meditation. Yet **Shashank Sharma** is confident he'll find nirvana with classy CLI tools.

OUR EXPERT

Shashank Sharma is in love with Go and Python, but Rust has been on his mind quite a bit lately.

The modern Linux kernel supports many different modules, which provide additional features and functionality. This could be support for USB devices, network protocols and other hardware.

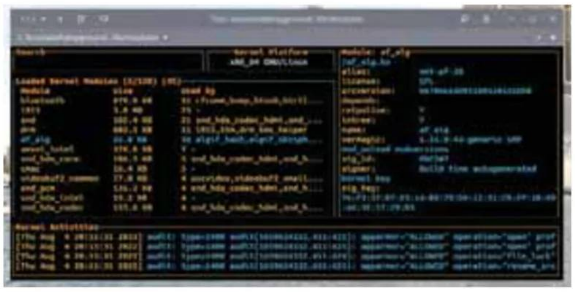
Only those elements that are of use for a large selection of users are an integral part of the Linux kernel, whereas other components are relegated to modules that users can enable at will. The advantage of working with modules to enable additional functionality is that you don't have to recompile the kernel, or even restart your machine to make use of the new feature set.

Released under the GPLv3 license, *Kmon* is a nifty tool designed to help you manage Linux kernel modules. Your Linux distribution features a number of different utilities that you can use to manage and monitor the different kernel modules. For instance, the `dmesg` command can be used to examine the kernel ring buffer and display the messages generated by the device drivers. The `uname -r` command can be used to list the version of the currently running kernel. The `lsmod` command can similarly be used to list all the kernel modules that are currently loaded.

Kmon feature a Terminal UI that incorporates all these utilities and more, to help you monitor the kernel and also load, unload or blacklist modules.

The obligatory installation

You'll find *Kmon* in the software repositories of some modern distributions such as Arch, Ubuntu and others. However, the featured version might be a bit dated. Fedora users can enable the COPR repository by running the `sudo dnf copr enable atim/kmon` command and then install *Kmon* with the `sudo dnf install kmon` command.



The LKM block is selected, where modules are sorted by number of dependents. Press Alt+d for a list of dependents of selected modules.

Because *Kmon* is written in Rust, one distro-neutral option is to use *Cargo* – the package manager for Rust utilities – to install the latest version. If you don't already have Rust and Cargo installed, you can use your distro's package manager to install these. The `sudo dnf install rust cargo` command will install these on RPM-based distributions.

Note that if you're running Ubuntu or Debian or one of their derivatives then you can similarly run the `sudo apt install cargo` command, which will also install the *Rustc* package.

Kmon requires the *XCB* package, but this isn't installed by default on most distributions. Depending on

» UNDERSTANDING KERNEL AND MODULES

In a nutshell, you can think of the Linux kernel as a resource manager that has the job of making sure that system resources such as processor, memory, disk, networking, etc. are adequately shared amongst multiple applications.

These days, many pieces of code are provided as modules, which aren't a part of the kernel but provide essential functionality. These can be device drivers, or support for certain file systems, network protocols, and so on. You can load or unload modules as required, without having to restart your machine with each such change.

You can use the `modinfo` command to display information about a kernel module, whereas the `modprobe` command can be used to load, unload or blacklist modules. The `lsmod` command can similarly be used to list currently loaded modules. The *Kmon* tool makes use of these utilities in the background to perform all its myriad functions.

We've extensively covered the Linux kernel in past issues. Look at **LXF215** where the cover feature celebrated 25 years of the Linux kernel. Cue issue **LXF280**, for the 30-year celebration. Both of these issues provide discussion on the kernel, it's development, the myriad components and how it all gels together into a thriving ecosystem.

your distribution, you might have to install the *libxcb-devel* package. With all that done, you can now run the `cargo install kmon` command to install the latest version of the utility.

Regardless of which of these methods you use, you won't find a man page for *Kmon* installed by default. This will have to be done manually. To begin, download the latest tarball from the project's GitHub releases page (**https://github.com/orhun/kmon/releases**) and extract the files:

```
$ tar xvf kmon-1.6.0-x86_64-unknown-linux-gnu.tar.gz
$ cd kmon-1.6.0
$ ls
CHANGELOG.md kmon LICENSE man README.md
$ cd man
$ ls
kmon.8
$ gzip kmon.8
$ cp kmon.8 /usr/share/man/man8
```

If you haven't already installed *Kmon* using any of the methods described above, you can also move the *kmon* script to a directory in the $PATH, such as **/usr/local/bin**, which would complete the installation:

```
$ cd kmon-1.6.0
$ ls
CHANGELOG.md kmon LICENSE man README.md
$ sudo cp kmon /usr/local/bin/
```

That's what we call a TUI

You can launch *Kmon* with the `kmon` command. The tool's interface features five different blocks: Search, Kernel Release, Module info, Loaded Kernel Modules and Kernel Activities. You can press the Tab key to cycle between Kernel Release, Kernel version and Kernel Platform.

All currently loaded modules are listed under the Loaded Kernel Modules (LKM) block. For each module, this block also provides its total size, as well as the number of other modules that require it.

The block that's currently in focus features a bold boundary line. You can switch between the different blocks using the right/left keys. You can similarly use the up/down arrow keys to navigate within the Loaded Kernel Module and the Module info block.

When you select a module, using the up/down arrow key, all pertinent information such as license, version, description and author is displayed in the Module info block.

If you prefer, you can even sort the different loaded modules by size, name or dependent modules with the `kmon sort <flag>` command:

Flag	Function
-n	Sort modules by name
-s	Sort modules by size
-d	Sort modules by their dependent modules

Run the `kmon` command first, without any options. Next, run the `kmon sort -n` and the `kmon sort -s` command to see the difference.

From any selected block, press 's' or Enter to move to the Search block. You can now enter the name of the module and all matching kernel modules will be listed in the LKM block. You must press the Tab key to move to the LKM block after searching for a module because pressing the down arrow key will switch to the Load module block.

To load a module, you can press the 'i' key, or switch to the Search module and press the down key. Type in the name of the module you wish to load, and hit Enter. You must provide the exact name of the module you wish to load, because the Load block can't be used to search for matching modules.

From the screenshots shown in this tutorial, you can see that we have a total of 120 loaded modules. You can also run the `lsmod | wc -l` command to confirm this. But how many modules are available in all, you are probably asking yourself? Run the `modprobe -c | wc -l` command to find out.

Kmon makes use of the default styles and colours of your terminal emulator window, but you can also define custom colours. Use the `kmon -a <colour name or hexcode>` command to define the accent colour, and `kmon -c <colour name or hexcode>` command to set the main colour.

Module options and sudo

When you select a module from the LKM list, you can press either the 'o' or 'm' key to open the options menu. This presents you with four options. Choose from Unload the module, Reload the module, Blacklist the module and finally, Show the dependent modules.

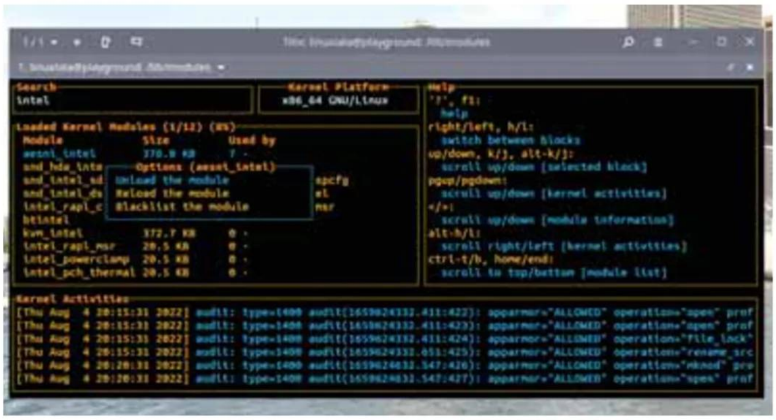
Each of these options can also be accessed using keyboard shortcuts. You can reload a module by pressing Ctrl+r key after selecting the module from LKM list. To similarly remove or unload a module, you can press the 'Backspace', '-' or 'u' keys. You can similarly press 'x' or 'b' to blacklist a module.

When you try any of these operations, *Kmon* will ask you confirm the step, and also show the relevant `modprobe` command that will be executed in the space where the Module info is usually displayed. You must press 'y' to confirm execution of the command. When you do so, *Kmon* will report an error that the specified module is currently in use and fail to execute the `modprobe` command.

This is because `modprobe` must be run as root or super user. When you're ready to use *Kmon* to load, unload or blacklist modules, you must run it with the `sudo kmon` command to be able to execute the different operations. LXF

QUICK TIP

If you dislike the order of the different blocks, press Ctrl+x to switch to one of three different layouts. You can also select a block and press Alt+e or Alt+s to expand or shrink the size of the block. Of course, you might also have to change the size of the terminal emulator window to get the most out of Kmon's interface.



You can access the help to view all available keyboard bindings by pressing '?'. The Help block replaces the Module information block.

» ENHANCE YOUR TERMINAL-FU Subscribe now at http://bit.ly/LinuxFormat

DROIDCAM

Credit: www.dev47apps.com

Set up your smartphone as a webcam

Nick Peers reveals how to convert your mobile devices into cameras you can use while chatting in Linux.

OUR EXPERT

Nick Peers has been playing around with computers for over 30 years, and has been dabbling with Linux for the best part of a decade.

Need a mobile internet protocol (IP) camera, but don't have the funds to purchase one? Want to make better use of your mobile devices – even the old ones that are gathering dust at the top of the cupboard in the spare room? Then *DroidCam* is the answer. This tool enables you to link your mobile to your PC so you can access its video camera as a webcam on your desktop. Whether using it to chat with others or even as an additional camera for video creatives (see the box, *facing page*), *DroidCam* has you covered.

Installation

DroidCam exists in two parts: the mobile app that you download from Google Play or Apple's App Store, and the client software on your PC. This installs a set of drivers – Video4Linux and ALSA loopback – and provides the connection to your mobile.

The 64-bit version of *DroidCam* is available to install as a binary. Open a Terminal window and then issue the following commands:

```
$ cd /tmp/
$ wget -O droidcam_latest.zip https://files.dev47apps.
net/linux/droidcam_1.8.2.zip
$ unzip droidcam_latest.zip -d droidcam
$ cd droidcam && sudo ./install-client
```

DroidCam uses a system tray indicator icon, which requires a dependency that's been removed from the latest builds of Ubuntu, Debian and Fedora. Ubuntu users can restore it with:

```
$ sudo apt update && sudo apt install libappindicator3-1
```

To finish the job, we need to install a loopback module. Just type the following:

```
$ sudo apt install linux-headers-`uname -r` gcc make
$ sudo ./install-video
$ lsmod | grep v4l2loopback_dc
```

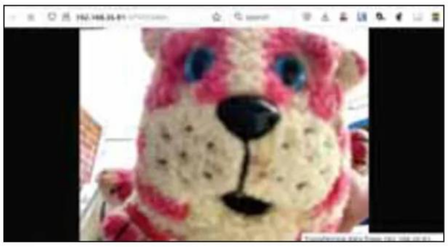
The final command should output confirmation that the loopback module is running.

Set up your webcam

Open *DroidCam* on your phone and run through the quick-start wizard. The developers recommend you only use it for video, so feel free to block access to the microphone when prompted. Make a note of the Wi-Fi IP and Port numbers, then open the PC version of *DroidCam* from the launcher. A pop-up window will appear. Leave Wi-Fi/LAN selected, enter the Wi-Fi IP address you recorded into the Phone IP box, verify the port numbers (4747) match, and finally click Connect.

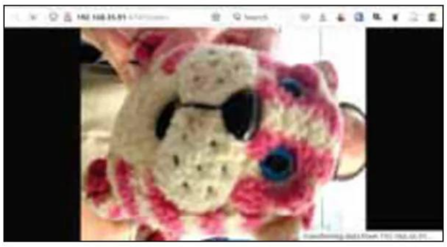
After a short pause, the *DroidCam* app should start streaming using your phone's rear camera – you'll find

USE YOUR PHONE AS AN IP CAMERA



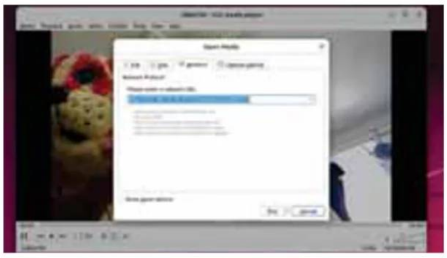
1 Connect via IP

Make sure *DroidCam* is open on your phone before opening your web browser and typing **http://<ip>:4747/video/** into the Address bar (substitute <ip> with the IP address displayed on the phone). You should see the video feed appear.



2 Correct aspect ratio

The video will display a mirror image of that shown on the phone. It also assumes that your mobile is set to display in landscape orientation, so if the image appears on its side, you'll need to flip your phone to landscape view.



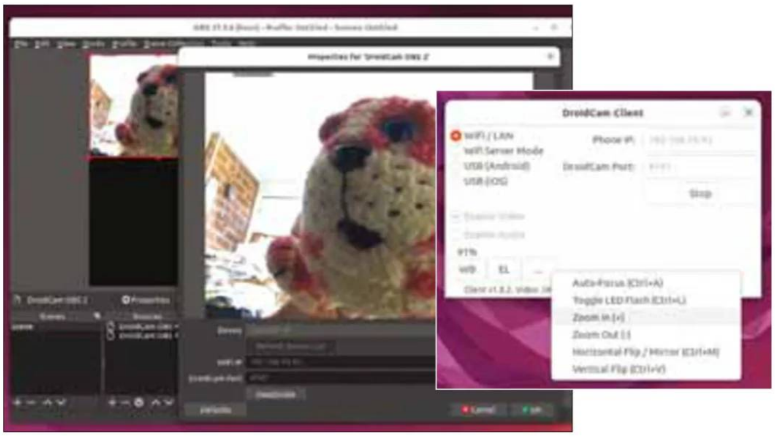
3 Desktop access

You can also access this feed via programs that support IP cameras – for example, *VLC*. In *VLC*, select Media>Open Network Stream. Type the same URL you entered in step one into the network URL box and click Play.

the control to switch cameras in the bottom left-hand corner of the screen along with EV (exposure) and AWB (auto white balance) buttons. These provide you with sliders to fine-tune the lighting and colour balance.

You'll also find similar controls – EV and AWB – in the Linux program, although they work in a slightly different way as you'll see shortly. For now, tap the Settings button in the mobile app to access controls such as preventing the screen from automatically dimming and changing the default port that *DroidCam* works on. You can also tweak camera settings from here, including quality, camera position (which camera to use by default), and a handy Limit FPS (frames per second) option to reduce bandwidth and power usage.

For most people, the defaults should be fine, but if you're connected to power or have a suitably fast connection then you may want to improve quality and FPS. Conversely, if you're on a slow connection, try reducing the quality.



Once installed, the Linux app is used to gain access to your phone camera. You can also tweak various settings. If you install the OBS Studio plugin then you can also use your mobile phones as additional cameras in your creative video projects.

Make a call

Once you've configured *DroidCam* on your phone and positioned the camera, turn your attention back to the Linux client. The WB button enables you to choose different preset lighting effects, with options ranging from incandescent and warm fluorescent to daylight and shade. EL is an option to lock the exposure control while the '...' button reveals further options, including a zoom control, support for flipping the screen and toggling both automatic focus and (if applicable) the LED light on your phone. Only the zoom controls work in the free version.

To hide the client from view, click the minimise button – if you've not already spotted it, you'll see the *DroidCam* icon in your menu bar – just click this and choose Show to bring the controls back.

Now switch to any Linux program with support for webcams. Here you should find that *DroidCam* is now an option alongside any other installed webcams. Explore the chat app's settings to see if you can set *DroidCam* as the default camera for messaging (Signal users should select File>Preferences>Calls, for example).

Free versus paid

DroidCam exists in two versions: free and Pro. If you're an iOS user, you simply upgrade to Pro for £5.99 within your existing *DroidCam* app; if you're an Android user, you'll need to purchase *DroidCamX* (£4.99) separately. The main advantage of upgrading to Pro is to unlock support for HD streaming. In addition, *DroidCam* iOS users will unlock access to all settings (basically those greyed out in the free version) as well as those on the PC client. After unlocking HD support, you'll need to update the loopback module to stream at a higher resolution. Visit **www.dev47apps.com/droidcam/linux** for instructions on how to do so.

Other connection types

If your Wi-Fi connection is too flaky, you can also connect to your mobile via USB. Connect the cable to your PC – iOS users will be prompted to trust the device – and then choose the appropriate USB option within the *DroidCam* app before clicking Connect. This provides a much more stable connection than Wi-Fi, but of course ties the mobile to the vicinity of your PC.

You can also use *DroidCam* as a straightforward IP camera. In this mode, you don't even need to install the software or have it running on your PC. The step-by-step guide (opposite) reveals all you need to know.

Finally, you'll notice a fourth connection option is available: Wi-Fi Server Mode. This only works with the *DroidCamX* Android app, and reverses the connection, enabling you to stream your PC's webcam (video only) to your phone. Visit **www.dev47apps.com/droidcam/connect** to find out more about these and other connection options. **LXF**

QUICK TIP

Should you wish to remove DroidCam at any point, do so with the following command: sudo /opt/droidcam-uninstall.

» LINK DROIDCAM TO OBS STUDIO

You can also use *DroidCam* to make it possible for you to deploy your mobile as a camera within *OBS Studio* (see **LXF269** for a tutorial on getting started with *OBS Studio*). First, download the OBS DroidCam plugin in zip format from **www.dev47apps.com/obs**. Once done, extract the files into a single folder inside your **Downloads** folder, then navigate to that folder in Terminal (such as **cd Downloads/plugin**) and issue the following command:

```
$ ./install.sh
```

This will create a **plugins** folder and copy the OBS Droidcam plugin into it. Now open *OBS Studio* and when you come to open a source, you'll see OBS Droidcam is an option. Select this, click OK and enter your mobile's Wi-Fi IP into the relevant box and click Activate (if the button isn't visible then you'll need to scroll down).

You can set up multiple mobiles as cameras – just add extra OBS Droidcam cameras and point them towards your other devices' IP addresses. Note, the iOS *DroidCam* app includes *OBS Studio* support; Android users will need to install a separate app (*DroidCam OBS*).

Free users are restricted to standard definition video – iOS users unlock HD support by buying a Pro licence for the main *DroidCam* app; Android users must pay £8.99 to unlock *DroidCam OBS*, but gain extra benefits including a remote camera control dock.

OPEN INDIANA

Credit: www.openindiana.org

Get to grips with Solaris

Linux veteran **Michael Reed** wonders if he can pass for a native in the weird and wonderful land of Solaris-derived OpenIndiana.

Michael Reed sometimes feels like he's OpenIndiana Jones when exploring an unfamiliar operating system.

QUICK TIP

OpenIndiana's website offers USB and DVD ISO installation images. The images are also offered as a text mode installer version and a minimal installer version. You need to know your way around a Solaris-style operating system to employ the latter option.

OpenIndiana is an operating system that aims to recreate the Solaris operating system. It uses some of the same source code as Solaris mixed with modern open source alternatives.

Solaris was a commercial, Unix-derived operating system offered by Sun Microsystems, and then Oracle, to power various workstations and servers. It is, itself, derived from various earlier Unixes such as System V, SunOS and BSD. This incarnation is a different lineage to Linux which is a 'ground up' Unix-like kernel created by an open source community that's been distributed with open source tools to form a working OS.

What OpenIndiana amounts to is a Unix-like OS that comes with an easy-to-use installer and a MATE desktop. This might sound like a Linux distribution, and there are many similarities, and nearly every major feature you would expect from a Linux distribution is present. However, much of it is implemented in a Solaris manner, and there's a lot to learn if Linux is the only Unix-like OS you've used. Let's see if a Linux-head can navigate the world of Solaris/OpenIndiana...

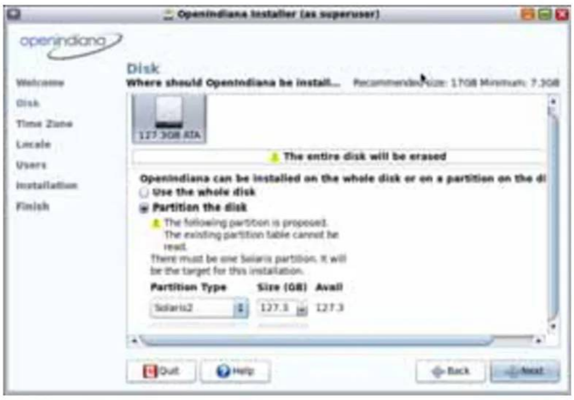
Sunny impressions

Given the lineage of this operating system, it's something of a surprise that the installation routine is so routine. It runs from a live GUI environment and resembles a typical Linux installation. One specifies the partitioning details along with the timezone and language before inputting details for the regular user and the root user. Click Finish, and after several minutes the user is invited to reboot the machine.

The default graphical environment, provided by MATE, didn't bowl us over with much in the way of personality. We're not putting down MATE, because it's a good choice for low-resource usage on a fully featured desktop, but in OpenIndiana it's configured to look exactly like Gnome 2.0. Yawn. Arguably, this combines a lowest common denominator usability with the familiarity of a standard business desktop that most people will be able to understand at a glance.

Along the top of the screen, we have an application launcher bar with Applications, Places and System pull-down menus. Unfortunately, these bars aren't searchable as they are with a standard MATE desktop. Further along, we have the usual control panel and status area, and along the bottom of the screen there's the standard task switcher.

Basic applications are present, such as the simple *Pluma GUI* text editor and the *Caja* file manager. Speaking of which, early on, we found out that our NAS, which uses the Samba protocol, was detected and usable. These smaller utilities sit alongside standard applications such as *Firefox*, *Thunderbird* and a VNC desktop sharing client. Beyond this, there are the usual cosmetic configuration applets that you'd expect from a MATE desktop.



Disk

Where should OpenIndiana be install... Recommended size: 17GB Minimum: 7.3GB

The entire disk will be erased

OpenIndiana can be installed on the whole disk or on a partition on the di

○ Use the whole disk

● Partition the disk

The following partition is proposed. The existing partition table cannot be read.

There must be one Solaris partition. It will be the target for this installation.

Partition Type	Size (GB)	Avail
Solaris2	127.3	127.3

Installation is surprisingly similar to that of a typical Linux live distribution. We chose the automatic partitioning option.

Hot architecture

Because they're both derived from Unix, Linux and OpenIndiana have similar architectures. On the lowest level we find the kernel and the hardware drivers. The Solaris kernel (and therefore the OpenIndiana one) was developed under a closed-source model for much of its life, and it evolved independently of the Linux one. Although a user doesn't interact with the kernel directly on a day-to-day basis, a stable and performant kernel is an essential component of an operating system. The OpenIndiana kernel offers features that are largely on a par with the Linux kernel, and we were pleased to see that OpenIndiana detected and utilised all four processor cores of our test system.

Like the Linux kernel, the OpenIndiana kernel is classed as monolithic, meaning that the drivers and many services operate inside the kernel. In the case of both Linux and OpenIndiana, the kernel itself provides a large proportion of the operating system's functionality.

Above the kernel level, we have the 'userland' which comprises the command line and GUI applications and tools and the graphical user interface itself. Because OpenIndiana is a Unix-type operating system, it can run a lot of the same software as better-known operating systems such as GNU/Linux and the BSD family of operating systems even though it's not directly compatible with them. OpenIndiana does have binary compatibility with Solaris, which probably only benefits the end user in some extremely niche server situations.

We started our investigation into OpenIndiana by looking at the command line. From the outset, this should be fairly familiar to most Linux users because the BASH shell is default and it's accessed via a terminal window. Commands that will be familiar to Linux users such as `ls`, `cat`, `mv` and `chmod` are present, and this shouldn't come as too much of a surprise because they were actually Unix commands that Linux inherited thanks to the GNU project, rather than the other way around. However, it's important to note that not every command is identical in how it works even though it may have the same name. For example `fdisk -l` doesn't list partitions like it does on a Linux system, even though an `fdisk` command does exist.

Once a new system has been installed, a system update is in order. Doing this threw up one of the first disappointments that we encountered because OpenIndiana doesn't currently come with a package management GUI. However, we were able to proceed by typing `sudo pkg image-update`. As with Linux, the `sudo` command carries out the command that follows it with root privileges. The packaging system of OpenIndiana is called the Image Packaging System, and *pkg* is the command line front end for working with it. `image-update` is the sub-command that causes *pkg* to update all packages on the system.

System updates and package installations were a slow process on OpenIndiana, although the system itself ran at a comparable speed to a modern Linux distribution in most other areas of operation.

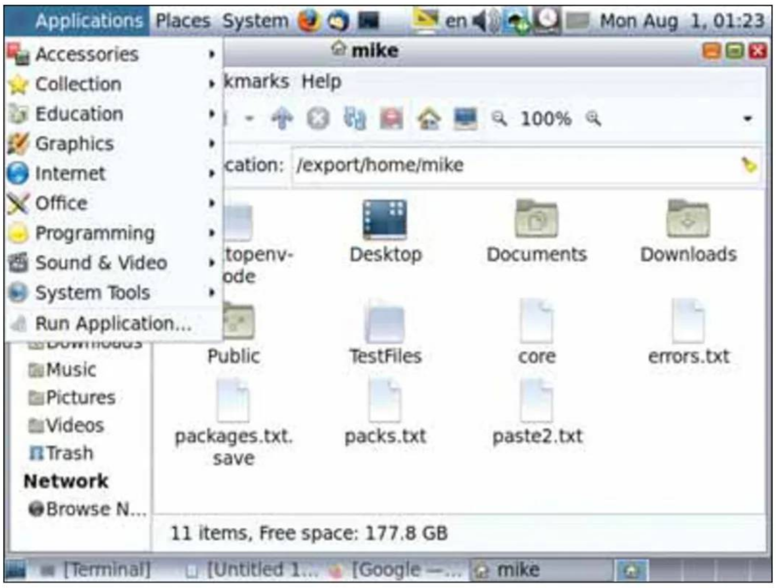
Eclipsed software

Having updated the system, we decided to add some familiar tools such as the *Nano* text-mode text editor. We first found out if it was available in the repository with the `pkg search nano`. We did like the logic of a hierarchy being employed as part of the package name. With the right command line switches, you can search within a category.

The command string to install Nano ended up being `sudo pkg install /editor/nano@6.3-2022.0.0.0` with `editor` being the category of package and the numbers being the version number of the package. In all fairness, some packages worked with a simple package name, but some required the fully qualified package name containing the category and the version number. Overall, it's a workable system.

Installing packages made us wonder what applications would be available, and in this area we were impressed because many well-known applications from the Linux world can be installed. By default, OpenIndiana comes with the *Firefox* web browser. This was up to date and worked as expected with no noticeable performance problems. However, *Chromium*, the open source version of Google's *Chrome* browser, wasn't available for installation.

Because the default desktop environment is MATE, the default text editor is *Pluma*. However, we ran into some problems using the program: it exhibited graphical bugs. For that reason, we added the lightweight *Geany* editor instead, and this worked perfectly. We were pleased to see that *LibreOffice* was available, so we installed it. The port worked as expected and was fully up to date. We also installed *GIMP*, but *Krita* (along with a lot of KDE applications) was absent. We installed *Inkscape*, and it ran perfectly well although it was an out-of-date version (0.92).

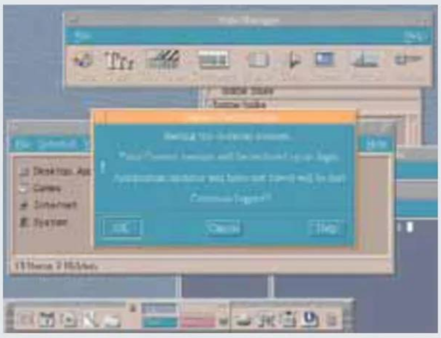


The user interface of OpenIndiana is MATE configured to look like Gnome 2.x, and features tools like the Caja file manager.

» THE USER INTERFACE

There was a time when Solaris, like many workstation and server Unixes, ran an interesting desktop environment called CDE (Common Desktop Environment). This is a desktop that received design input from many of the heavyweights of the workstation scene of the time such as Hewlett Packard and IBM. By the late-1990s, Sun had decided to abandon CDE for a more standardised Unix desktop called the Java Desktop System, which was largely a rebadged version of Gnome. In this spirit, OpenIndiana uses the MATE desktop environment, but it's configured to look similar to Gnome 2.x. It's not an exciting-looking environment. We investigated the options to either modernise things with KDE or to revive the classic look and feel of a mid-1990s Unix workstation with CDE, but we struck out in both cases: the package and source code options were uninstallable on modern OpenIndiana.

However, CDE did compile and install on Ubuntu Linux when we tried it. We did this by following the official instructions at **https://sourceforge.net/projects/cdesktopenv/** and this proved a great desktop to explore. It's just a shame that we couldn't pair the CDE user experience with OpenIndiana for a truer '90s era Unix experience like being back at Uni!



The Common Desktop Environment, the face of many Unix workstations in the 1990s. It's now open source and available for Linux.

QUICK TIP

As per the official instructions (www.openindiana.org/br/packages) we added the 'hipster-encumbered' publisher (equivalent of a Linux repository) for patent-encumbered multimedia codecs and the 'Spec Files Extra' to add some extra ports of modern applications.

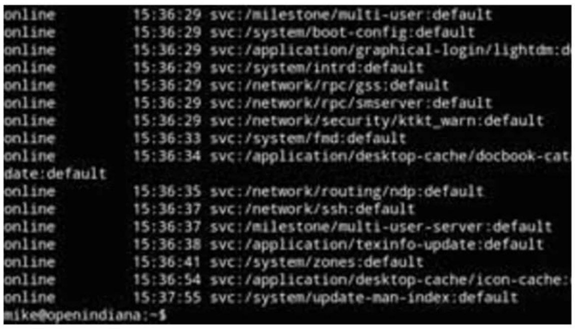
When it came to media players, our usual go-to, *VLC*, was listed in a search but proved uninstallable due to unresolvable dependency issues. *FFmpeg*-based player *FFplay* installed but wouldn't run due to being unable to find libraries that it needed. In the end we got the rather basic *Totem* player working. This played the video and music files that we threw at it. *Audacity* is normally our go-to audio editor and it was installable and perfectly usable, although it was an outdated 2.x version. At the time of writing, not every Linux distribution has the 3.x series available, but in the case of Linux it's possible to install the AppImage version, and OpenIndiana can't run AppImages.

In summary, we felt that the selection of installable applications was reasonably impressive, probably due to the architectural similarities between Linux, Solaris and therefore OpenIndiana. A lot of the heavy-hitters such as *Firefox* and *LibreOffice* were present, and most software areas have a workable application that can be installed. However, there are plenty of situations where you might find yourself having to make-do rather than using your first choice of application. There's some binary compatibility with Solaris, but this seems to be mostly for niche server applications.

Sun zones…

OpenIndiana, like its parent Solaris, is a currently maintained server operating system, and it's inherited some advanced features from that parent. Zones, containers that are isolated from the rest of the system, are an example of such a feature.

We'll give you a flavour of what it takes to create an example zone. You create a zone by first creating its configuration with the `zonecfg` command. Hit the official docs for how to configure a zone (**http://docs.openindiana.org/handbook/systems-administration/#zones**). Briefly, it involves typing about 10 commands into an inline editor (that reminded us of manually setting up a MySQL database). This is where you set up the virtual network interface and specify where, within the file system, the storage for the zone will exist as a bare minimum.



```
online          15:36:29 svc:/milestone/multi-user:default
online          15:36:29 svc:/system/boot-config:default
online          15:36:29 svc:/application/graphical-login/lightdm:d
online          15:36:29 svc:/system/intrd:default
online          15:36:29 svc:/network/rpc/gss:default
online          15:36:29 svc:/network/rpc/smserver:default
online          15:36:29 svc:/network/security/ktkt_warn:default
online          15:36:33 svc:/system/fmd:default
online          15:36:34 svc:/application/desktop-cache/docbook-cat
date:default
online          15:36:35 svc:/network/routing/ndp:default
online          15:36:37 svc:/network/ssh:default
online          15:36:37 svc:/milestone/multi-user-server:default
online          15:36:38 svc:/application/texinfo-update:default
online          15:36:41 svc:/system/zones:default
online          15:36:54 svc:/application/desktop-cache/icon-cache:
online          15:37:55 svc:/system/update-man-index:default
mike@openindiana:~$
```

OpenIndiana uses SMF (Service Management Facility) that it inherited from Solaris to manage services. Here, we're SSHing from a Linux terminal.

To make zones work, we had to carry out a bit of ZFS magic at the command line. Although we only had to do this once, the procedure wasn't mentioned in the official Hipster Handbook (see box, *below*), and we consider this to be typical of the type of oversight that you're likely to run into when dealing with the official documentation.

Once the zone has been configured, install it: `sudo zoneadm -z [name of zone] install`. When you do this, the package management system will add a minimalist set of packages to the zone, and like everything involving package management on this system, it takes quite a long time. However, once the zone is configured and installed, accessing it is fairly simple. Boot the zone: `sudo zoneadm -z [name of zone] boot` and log into it with `sudo zlogin [name of zone]`.

What you have at this point is an isolated zone that acts as though it's an independent OpenIndiana machine and is accessible via the command line. It's ideal for installing servers. It's not quite the same environment as a Docker container because the zone is based on the underlying operating system, and we'll leave it to the reader to decide if it's an easier workflow.

We've only scratched the surface of what zones can do, and it's possible for zones to take advantage of ZFS features such as snapshotting and rollbacks. It's also possible to fix it so that package installations and upgrades are carried over to zones when they are carried out in the general system.

Service management

OpenIndiana is a server-orientated OS, so you'll probably spend some time managing services. We found that the overall procedure is similar to Linux.

We wanted to configure SSH so that we could access the OpenIndiana command line from a Linux desktop,

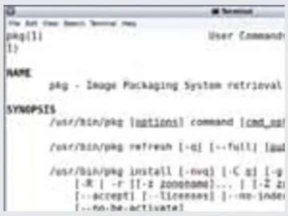
» DOCUMENTATION AND SUPPORT

OpenIndiana is a more niche operating system than Linux, so there simply isn't as much help available for it. To get by, you'll have to rely on the official documentation, documentation for Solaris and general documentation for other Unix-derived operating systems such as Linux and the BSD family.

The currently maintained version of OpenIndiana is called Hipster, and the *The Hipster Handbook* (**http://docs.openindiana.org/handbook/getting-started**) should be your first port of call when trying to figure out how to do things on it. This is divided into sections, but some of it is either out of date or incomplete. In particular, outdated documentation gives examples that use the `pfexec` command, but as far as we could tell this has been deprecated in favour of `sudo` for administrator-level commands.

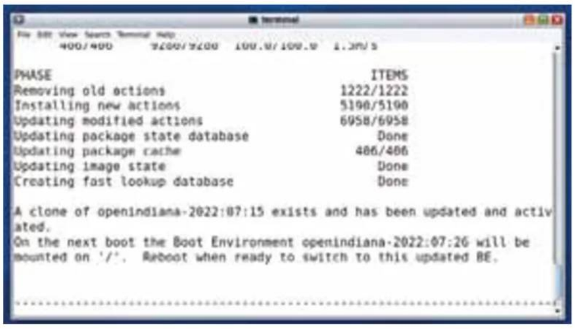
OpenIndiana is actually a distribution based on another operating system called illumos, the original open source fork of Solaris. When searching online, some useful information that applies to OpenIndiana can be found by searching for illumos documentation. For example, the illumos subreddit (**www.reddit.com/r/illumos**) is the most active forum we were able to find concerning OpenIndiana questions.

Fortunately, as a typical Unix-type operating system, good old-fashioned man pages are present, and these were extensive and comprehensive.



```
pkg(1)                                  User Command
1)

NAME
       pkg - Image Packaging System retrieval

SYNOPSIS
       /usr/bin/pkg [options] command [cmd_op

       /usr/bin/pkg refresh [-q] [--full] [pu

       /usr/bin/pkg install [-nvq] [-C n] [-g
           [-R | -r [[-z zonename]... | [-Z z
           [--accept] [--licenses] [--no-inde
           [--no-be-activate]
```

Man pages are a sight for sore eyes when navigating an unfamiliar operating system.



```
     406/406       9260/9260   100.0/100.0   1.3M/s

PHASE                                              ITEMS
Removing old actions                           1222/1222
Installing new actions                         5190/5190
Updating modified actions                      6958/6958
Updating package state database                     Done
Updating package cache                           406/406
Updating image state                                Done
Creating fast lookup database                       Done

A clone of openindiana-2022:07:15 exists and has been updated and activ
ated.
On the next boot the Boot Environment openindiana-2022:07:26 will be
mounted on '/'.  Reboot when ready to switch to this updated BE.
```

A complete system update using the pkg command. Package management was a little slow compared to a typical Linux distro.

and it served as a good example of the similarities and differences between OpenIndiana and Linux when it comes to administration procedures. We already had a working and tested SSH client on our Linux box.

`svcs` is the command that lists all active services on the system. It produces quite a lot of output, so we used `svcs | grep ssh` to search for the SSH server and that produced the following output:

```
online       12:36:33 svc:/network/ssh:default
```

The information presented is the service status, the service start time and the name of the service. As with package naming, we like that service names have a hierarchy. This means that you can search for a particular category of service.

Because the SSH service was already enabled, we discovered the IP address of the OpenIndiana machine with `ifconfig` in the same way we would in Linux. Then it was simply a matter of typing `ssh <username>@<OI ip address>` into a Linux terminal in order to log in.

Our next task was to disable insecure password logins, and most of the process was the same as it would have been with a Linux machine. While we could still login using our password, we executed `ssh-copy-id <username>@<OI ip address>` on our Linux box.

We edited 'PasswordAuthentication' to 'no' and 'ChallengeResponseAuthentication' to 'yes' in the **/etc/ssh/sshd_config** configuration file using the *Nano* text editor. We then generated the SSH Keypair with `ssh-keygen -t rsa` command on the OpenIndiana box.

A difference with Linux is that the command sequence that we used to restart the SSH server was `sudo svcadm restart network/ssh`, with `svcadm` being the command used to administer services on OpenIndiana. Following all of this we could then log in, securely, without a password by typing `ssh <username>@<OI ip address>` into a Linux terminal.

Blinded by the sun?

OpenIndiana offers many of the same facilities as Linux, even if some of them come under different names and are operated by different commands. In most areas, performance was reasonably good compared to Linux running on the same hardware, although system start up and package installation and maintenance were slow. There was a great deal of uniformity, compared with the variety of working methods exhibited by the multitude of Linux distributions, when it came to areas such as admin commands.

Application support is good but not brilliant. A lot of heavy-hitters are present such as *Firefox* and *LibreOffice*. However, there were also a lot of notable absences compared to a Linux desktop (such as many KDE/QT applications) and some applications were out of date. Generally, most application types within the business computing side of things are represented, but you might not be able to have your first choice in areas such as web browsing.

The desktop environment is functional but basic, and there were few working options to replace it with something more modern. There is support for some hardware graphics acceleration, but at present there isn't any support for 3D graphics acceleration.

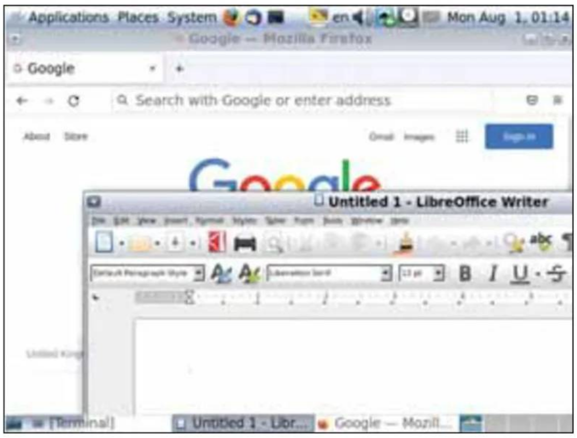
Considering its viability as a daily-driver, OpenIndiana fares reasonably well but the main thing working against it is that everything that it offers is available on the Linux desktop, and the Linux desktop has far more options for customisation and application choice. Documentation is far more extensive and much more likely to be up to date in the Linux world as well. Other alternative operating systems like Haiku (the recreation of BeOS) or RISC OS offer a completely different way of doing things with reasonable application support along with excellent performance and ease of use on the desktop. However, if we're making comparisons, neither of those two offer fully functional web browsing on the same level as *Firefox* did under OpenIndiana. Nor would either of them be a good choice as a general-purpose server.

OpenIndiana makes more sense as a server OS. We liked the idea of zones because they're more closely tied to the host system than Docker containers. However, Docker or LXC containers could be used to accomplish much the same things, even if they work slightly differently. It's good that OpenSolaris has a powerful filesystem, ZFS, as its base, and there might have been a time when we would have eyed it up enviously, but Btrfs and XFS can do most of the same things, and Btrfs is beginning to enter the mainstream as a default filing system on distributions such as openSUSE.

It was fascinating to examine a Unix offshoot OS that's largely evolved independently of Linux. It's a highly competent operating system, particularly when employed as a server. However, it's very similar to Linux, and Linux offers quite a bit more choice, particularly when used as a desktop OS. **LXF**

QUICK TIP

OpenIndiana, like Solaris, uses a filesystem called ZFS, and this has most of the features we've come to expect from a modern file system. You could run OpenZFS on Linux, but a better supported system such as Btrfs might be a better choice.



Fully functional applications such as Firefox and LibreOffice cover a lot of bases for potential office desktop use.

» GET MORE OUT-OF-DATE STUFF! Subscribe now at http://bit.ly/LinuxFormat

BACK ISSUES » MISSED ONE?

ISSUE 293
September 2022

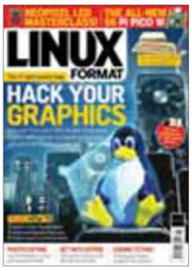
Product code:
LXFDB0293

LINUX FORMAT
GET INTO CODING!

In the magazine
We explain coding principles, what editor to use, how Git can help, and help you take your first steps writing your own code and controlling physical Pi projects. We review five of the best 3D printers, put together a CCTV system, assess a recreation of the BeOS operating system, and show you how to create custom fonts. There's also tutorials on Web 3.0, screencasting and cloud hosting.

ISSUE 292
Summer 2022

Product code:
LXFDB0292



In the magazine
We demystify the world of graphics on Linux with our in-depth feature. Elsewhere, we edit photos with *Photoflare*, explore RISC OS, try out dev environments using Gitpod, control multiple LEDs around the home with a Raspberry Pi, and add gamepad support to *Tetris*. We also assess five software whiteboards for collaborative teaching, and talk to the talented coder behind the *Amiberry* emulator.

ISSUE 291
August 2022

Product code:
LXFDB0291

LINUX FORMAT
FAST VIRTUAL MACHINES

In the magazine
Learn how to make your virtual machine go faster and run a virtual GPU. We show you how to compose music using algorithms, back up and copy entire hard drives, and advise on best password practice. In the market for a new distro for your Pi? We test five of them, plus check out the story behind Bodhi Linux, emulate the first portable computer, and take a deep dive into World Web 3.

ISSUE 290
July 2022

Product code:
LXFDB0290

LINUX FORMAT
HACKER'S TOOLKIT 2022

In the magazine
Expand your skill-set by exploring the formidable hacking tools within Parrot Security OS. Get the best fully-featured CMS for to host your web content on. We also explore sensors for the Pi, control Docker over the internet, emulate the Commodore 16, get to grips with FreeCAD Sketcher's Workbench, and learn the history of Pop!_OS from System76 developer Michael Murphy.

ISSUE 289
June 2022

Product code:
LXFDB0289

LINUX FORMAT
BULLET-PROOF UBUNTU 22.04

In the magazine
Discover what's new in the latest version of Ubuntu with our guide. We bring you a hands-on review of Valve's Steam Deck, rate five open source app stores, explain how to deep-clean your hard drives, get more from the Windows system layer *Wine*, run virtual machines with ease and emulate the Amstrad PCW. Plus we test your reflexes with a Python-based reaction game!

ISSUE 288
May 2022

Product code:
LXFDB0288

LINUX FORMAT
ESSENTIAL Pi PROJECTS

In the magazine
From flashing lights to image recognition and even an aircraft tracking system – we show you how to get more from your Raspberry Pi. Elsewhere, we transform photos and video with the G'MIC plugin, emulate the Acorn Archimedes, process satellite imagery, find out how practical it is to use a Pi as your daily driver, and code memory-secure systems in the programming language Rust.

To order, visit www.magazinesdirect.com

Select Single Issues from the tab menu, then select Linux Format.

Or call the back issues hotline on **0330 333 1113**
or **+44 (0)330 333 1113** for overseas orders.

Quote the Product code shown above and have your credit or debit card details ready

USA? EU? THE MOON*?

UK readers turn to p16

SUBSCRIBE!

Don't wait for the latest issue to reach your local store – subscribe today and let *Linux Format* fly straight to you. Faster, cheaper and with DRM-free archive access!

CODE & BUILD Pi PROJECTS!
+
Pi-RUN CCTV PROTECTION
LINUX FORMAT
The #1 open source mag
Create & host cloud services for FREE!
GET INTO CODING!
Everything you need to start writing apps, discover IDEs, get into Git and create code today
PLUS: HOW TO
» Discover modern BeOS
» Record and edit screencasts
» Administer servers like a pro
We test-drive the five best 3D Printers for makers & education
INSIDE WEB 3.0
Start coding for the next web revolution
ENHANCED FONTS
Use open source ones or design your own!
RUST NETWORKS
Write basic server and client applications

3 GREAT WAYS TO SUBSCRIBE

Print, digital-only, and print+digital bundles!

» **USA**
From $132
For 13 issues

» **REST OF THE WORLD**
From $132
For 13 issues

» **EUROPE**
From €100
For 13 issues

IT'S EASY TO SUBSCRIBE!

Visit **www.magazinesdirect.com/linux-format**

Call **+44 0330 333 1113**

Lines open Monday-Friday, 9am-5pm, UK time.

*We don't actually deliver to the Moon. Yet.

CURA

Credit: https://github.com/Ultimaker/Cura

Part One! Don't miss next issue! Subscribe on page 16

Getting started with Cura and 3D printing

Not getting the quality of prints you expected from your 3D printer? **Ali Jennings** shows you the essential steps the manuals often miss.

OUR EXPERT

Alastair Jennings has been obsessed with 3D printing for well over a decade. And it all started with a large, heavy, noisy and frankly terrible 3D printer.

Just purchased your new feature-packed 3D printer, but failing to reproduce the high-quality prints you've seen advertised? Don't worry, this is quite common, and even though the sample print that shipped with the machine may have printed perfectly well when it comes to printing your own models, things can start to go a little astray.

If you've already extracted a decent print from your machine, you're well on the way to a happy 3D printed life. This tutorial will still point out a few tips and tricks to help that continue. If you've roughly set up the printer and prints are a little hit and miss then we'll delve into *Cura's* and the hardware settings you need to check.

Don't worry if 3D printing is new to you, and your machine has arrived or is about to. We'll delve into how to check your machine, set up *Cura* and look at some of the common settings that many manufacturers often skip over.

That leaves the final group those of you who have the printer but as yet have progressed no further than getting it to make awful sounds every time you try to print. By the end of this tutorial series you should be 3D printing like a pro.

Read that fine manual!

The first step is to read the manual, if instructions come with your printer. As you flick through the pages, pay special attention to the cable or wire routing. How this is positioned and fixed to the frame can make a huge difference to the operation of the machine. It can be the case that a wire or cable is causing undue stress or friction on the motors. This in turn affects the speed of motion, directly reducing the print quality.

Check the space required for your printer. It may look quite compact, but once it's switched on and come to life, 3D printers have a habit of expanding. Does the printer have enough space on the work surface in order for the build platform to move back and forth freely? Is there room for a filament reel? It's common to clear a space on a worksurface, only to find that a cupboard is blocking the filament holder. 3D printers, however small, take up more space than you'd imagine.

Before you start, also think about where the printer will live. Filaments such as ABS are oil-based and give off fumes that are best avoided. High-temperature filaments also need a more controlled environment to stop them from warping, while PLA needs a cool room to ensure it sets quickly. Think about where to put your printer, and consider its position all year round. The one thing a 3D printer hates more than anything else is moisture. Many popular filaments, such as PLA, are porous and absorb at speed, rendering them useless. Likewise, direct contact with sunlight will degrade materials, making them brittle and unusable.

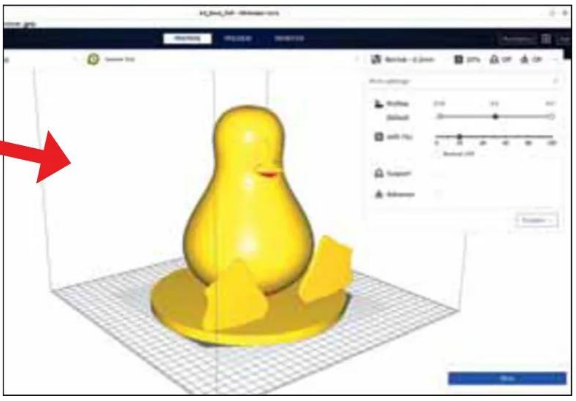
Once you've found a home for your 3D printer, give everything a final check. Make sure all its nuts and bolts are tight and then check each axis to ensure that any bushes or bearings are correctly set and running free. Many printers now offer adjustable bearings and runners so that you can fine-tune the motion.

Now check the belt tensions. Your printer will usually have two belts, one on the X- and the other on the Y-axis.

QUICK TIP

Don't leave your 3D filament lying around. Its filament will be one of the biggest influences on your 3D printer's print quality. Once you open a new pack of filament, keep it in a dry and dark place when not in use, preferably in a sealed container and at a constant temperature.

When it comes to successful prints a little time fine-tuning your printer and CURA settings can go a long way to better, more productive prints.



robe Z Offset:
-1.100

Carefully watching the quality of the first layer and micro-adjusting the Z-axis height will make a significant difference to the quality of your prints.

To check these, push the two sides together. You should feel a little firm resistance but not too much. If the two parts of the belt touch easily then tighten, but if they take some effort to touch, loosen.

Before you switch on, check that the build platform is positioned correctly and flat. There's often a screw edge or clip on which the platform can get caught. If this is the case, the rest of the setup process will go straight out the window.

In the past, most machines required you to level the print platform manually. Today most printers feature an auto bed levelling feature, so before playing around with the print platform, it is worth printing a sample print, one is usually provided.

Turning on the printer

On starting up the machine, a self-calibration process usually starts with the tool head dipping down and touching the print platform before moving on. This process varies with each machine, but is now a standard feature. While the feature usually works fine, it often needs a bit of a tweak to get it right. We'll take a look at that in a bit.

Most manufacturers will supply you with a test 3D model on a USB key, MicroSD or SD card. Pop this into your machine and hit Print. As the printer starts, watch carefully for any judders or movement in the frame and rectify these when the print finishes.

As this first print starts, look at how the first layer is laid down. The quality of this layer will dictate the quality of the rest of the print. The adjustments we'll make to the printer will all be based on the first layer.

Hopefully, you'll have instant success with the first layer, and you'll be able to move on and start printing your models. However, if your model failed to stick to the print platform, this could be down to several factors.

The first thing to do is to check what's known as the Z-axis height. This is simply the distance of the nozzle from the build platform. While the auto bed levelling usually does a good job of correctly gauging this distance across the print platform, it also often needs a tweak. Look at that first layer and the filament lines as they're laid down. If the lines are faint, the nozzle is too close, probably scraping the print platform, so the nozzle needs to be raised. If the lines feature small gaps between each row, the nozzle is too high and needs to be lowered.

A correct height nozzle will reveal a precise line and take on the look as though the model's been gently pressed into the build platform. Looking at the lines side-by-side, you should also see that they are neat and close together.

Once the mechanical fine tuning is completed, we can look at setting up the slicer software. At this point, it's worth going through the 3D print workflow. 3D printers work by printing out sliced 3D models; the models are created in a 3D application such as *Blender* or *Cinema4D* and then exported as a dxf, obj or STL file. These 3D files are then imported into slicing applications such as *Cura* or *Prusa Slicer*. These free applications can be configured with almost any 3D printer presently on the market or invented at home.

Once the model is in the slicer software, you can select the print quality options and settings. The main option is quality; this essentially relates to the resolution of the print, the print speed, outer wall thickness and infill. Once the software calculates all these parameters, the model is ready to export to the 3D printer.

That exported file is a set of commands that plots a path layer-by-layer for the printhead to follow. As each layer is completed, the next layer is added until the final 3D print is completed. This technique is relatively simple but has a few issues regarding fine details and objects with nothing to support them. We'll again cover that later.

The slicing software links the 3D model with the coordinates for the 3D printer to do its work. Because every 3D printer is different, you must set up the options within your program of choice before you send

QUICK TIP

As the weather cools, using your printer in the shed, garage, or outdoor workshop can cause printer problems because the filament sets too quickly. Stick on a heater before printing and ensure the room's ambient temperature is above 15ºC.

» SETTING THE Z-AXIS

Although most 3D printers will come calibrated before shipping, the Z-axis or height of the tool will have a significant effect on the quality of prints. Just a touch too high and the print is unlikely to stick; too low and the nozzle will scrape the print bed and could ultimately block the nozzle.

Before fine-tuning the Z-axis settings, run through the printer's standard setup process. If you're lucky, everything will start to print perfectly the first time. However, it's more likely that some adjustment will be required.

Start by printing a calibration cube (a simple cube) and watch how the first layer is laid down. You're looking for neat lines slightly pressed into the build platform. If the lines are faint, the nozzle is too close to the platform; if there are gaps between the lines – however faint – then the nozzle is too high.

On almost all modern 3D printers, you'll be able to adjust the Z-axis quickly and often while printing to see the effect. Look at the Lulzbot SideKick 747, and in the Calibration menu options you'll find Z-Offset. Adjust this to change the nozzle height: '-' it usually takes the nozzle closer while '+' draws it further away.

»

QUICK TIP

ABS and other high-temperature 3D print filaments need a constant and warm atmosphere to print at maximum quality. If you want to print with these materials rather than PETG or PLA, either build an enclosure around the 3D printer to keep the warmth in or increase the room's ambient temperature.

them across to the printer. In this tutorial, we'll look at *Cura*, which is freely downloadable from: **https://ultimaker.com/software/ultimaker-cura**.

Once downloaded and installed, *Cura* asks you to add a printer. You'll need to dig out your spec sheet for your model. If you're lucky, the details will already be in the *Cura* Database and download automatically; if not, you'll need to input them yourself; we've run through this in the three-step walkthrough (*facing page*).

Once done, you're essentially finished with the *Cura* setup and should be ready to print. However, the next stage is to test your setup. This takes a bit of time but will ensure happy printing in the future. These tests check the configuration and, ultimately, print quality.

The first print is the Calibration Squares. You'll find various options, such as **www.thingiverse.com/thing:4894942**. This grid prints out the first layer across patches of the print platform. You can then check to see if the print platform is at the correct height in all areas of the print platform. While not all printers will enable you to fine-tune the settings for various parts of the bed, many models like the Original PRUSA M3S do. The platform level can be slightly adjusted if the layer isn't being laid down correctly in the top right corner. On other printers it may be a more manual process of turning levelling knobs or even slipping a small bit of card under the print platform to raise it slightly.

Blocked nozzles are a pain, but using the hot or cold pull method will usually release all but the most stubborn burnt-in filament.

Once the grid is well laid down, you can move on to print quality. The first model to look at is 3D Benchy (**www.thingiverse.com/thing:763622**). This stress test is one of the most popular, with the small boat quickly showing off any flaws and details.

It's a good one to run when you want to test the different quality settings and the time it takes to print. For example, on the Lulzbot TAZ SideKick 747, a quick print takes 41 minutes, a standard print takes one hour and 5 minutes, and high quality takes one hour 45 minutes. There's a big difference between each print, so a test like this is good to run so you know what setting you'll need for different print types. It's also worth noting that while draft or quick, low-resolution prints might not have the print finish of the high-quality, slower prints, they're general stronger, which makes them a better option for functional parts.

The final print to run is the AutoDesk Kickstarter model Benchmark (**https://github.com/kickstarter/kickstarter-autodesk-3d/tree/master/FDM-protocol**). It puts printers through their paces, testing dimensional accuracy, overhang, negative features and control. It also shows XY Resonance and Z-axis alignment, which you can use to fine-tune your machine.

Printing models

Once those three simple models and the test model from the manufacturer have been printed, you're then ready to print your 3D models. After you've loaded your models, this is where is a little knowledge of *Cura* and some of those settings are helpful. To make things easy, *Cura* gives you an easy-to-use interface. Load in your model by clicking the folder icon, then select the first option under Print Settings: Quality. For the example printer, the Extra Coarse option is a low-resolution 0.6mm layer height print, whereas the Extra Fine at 0.06mm layer height is a high-resolution quality print. If you're unsure, then Normal is a good place to start.

After that, set the other options. You should pay attention to Infill, Support and Adhesion. There are plenty of other options that you can investigate as you become more familiar with the print process.

Infill is usually set to grid as standard; this is a good all-around infill pattern that takes up the room inside your print so that it isn't hollow and weak, or solid and

» HOUSEKEEPING

3D printers are pieces of precision equipment, and the better you look after them, the better the prints you can expect from them. One of the most important processes of 3D printing is cleaning – making sure everything is clean, clear and ready for the next print.

After every print, check the print base is cleaned and clear, remove any glue if using a glass plate and scrape it clean. If you're using a flexible steel plate, make sure all the bits of plastic have been removed and give it a quick wipe with some 70 per cent isopropyl alcohol on a lint-free cloth. This one step is going to help you avoid so many misprints!

Next, make sure there are no threads of plastic blocking belts, cogs or other moving parts. It's amazing how many small fragments of plastic quickly appear once you start 3D printing. A quick vacuum can easily clear other dust and particles. However, be careful not to knock anything delicate.

Then make a quick visual inspection and check the belt tension. A good check is to pinch the two belts together – they should offer some resistance before they touch. If the belts feel slack, tighten them; if they're too tight, loosen.

Finally, check the nozzle. The better the condition, the better your prints will be. First clear it of plastic; warming the nozzle to 140°C for PLA and 180°C for ABS will help, but don't touch the nozzle with your fingers. Then use a small wire brush to remove any stuck filament that might catch on new filament during the next print.

» UNJAMMING A NOZZLE

A jammed nozzle is one of the most infuriating issues with any 3D printer. The most common reasons for this are that a cheap filament has been used, and the consistency of the material is out. This results in the gears that feed the filament through the system slipping, which will ultimately cause the filament to wear. Cheaper filament can vary in diameter, again causing the gears to slip or the nozzle to be at the incorrect heat for the filament to flows through evenly.

Another issue can be that the material has degraded. It may have been open for a while and become brittle or absorbed moisture. When filament becomes brittle, it can break at any point, but that point is often just after it's entered the extruder gearing. The pressure of the gears on the filament is enough to cause a break, and often requires a little dismantling of the tool head to clear the problem.

Finally, the worst blockage is often caused by expanding filament. This happens when the filament has absorbed moisture, and as it enters the hot end it expands and blocks. First unload the filament and check the position of the filament's end. If you see it sticking out of the extruder, preheat the nozzle and try using tweezers to extract the filament. If this doesn't work, take another length of filament, preheat the nozzle to the maximum filament temperature, and push the new filament in, hopefully unblocking the nozzle. You can buy thick metal needles for this same job.

If that still fails then try the hot and cold pull methods. Hot pull involves heating the nozzle, pushing the filament in until you feel resistance for a second or two, then pulling it quickly out, cutting off the end and repeating until you see filament extrude from the nozzle.

The cold pull is much the same: heat the nozzle, insert the filament, let the nozzle cool slightly, and then pull.

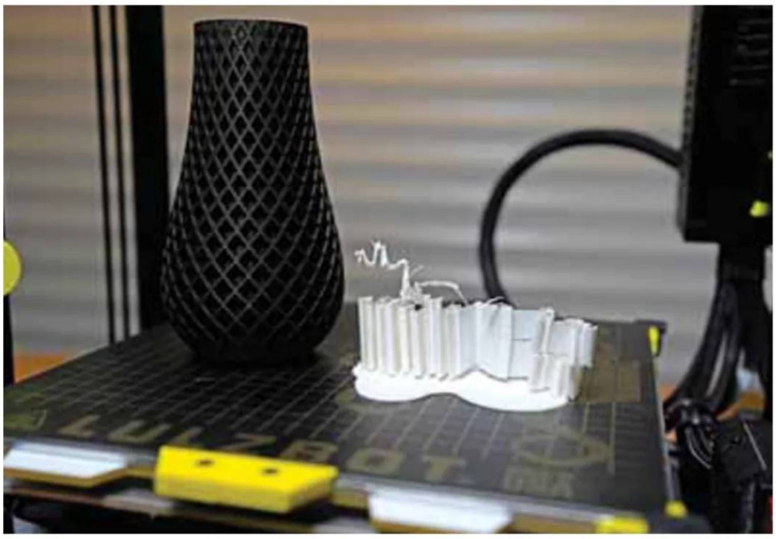
Another popular option is to use a fine needle. Insert the needle from the outside; you'll need to heat the nozzle to melt the filament inside.

wasteful of plastic. The grid pattern is a standard pattern, but gyroid is another that's now more commonly used. The usual infill amount is set to around 20 per cent. Any less, and you tend to get weak models; higher percentages are good if you need a stronger print, but does of course use more filament.

Other essentials are Support and Adhesion. The former hold up parts of the model with no underlying structure. For many items, supports are essential, and there are plenty of additional options for them. The issue with supports is that they're attached to the model and base, so they must be removed when the print has finished. For many models, it's worth switching these off if you can see that there are no overhangs likely to cause trouble.

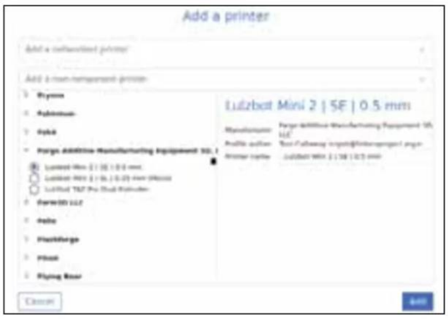
Adhesion is essentially a platform printed under the model. These also need to be removed from the finished model. They help with the overall print quality.

Once everything is set, you're free to print directly from the computer or via a USB drive or SD card. LXF



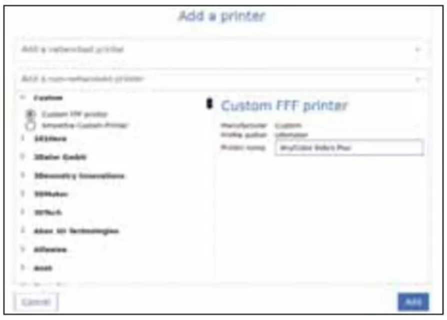
Spending a little extra time setting up your printer and making sure everything remains clean can make a major difference to the success of your prints.

SET UP YOUR PRINTER WITH CURA



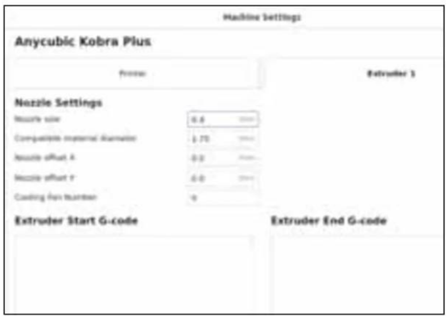
1 Predefined printer

Upon starting *Cura*, you'll be asked if you want to set up a new printer. By default, *Cura* has a good range of printers installed, and this list is updated regularly. Click your manufacturer and then your printer model,. The parameters for your printer will be installed, and you're set to go.



2 Defining your printer

Start *Cura* and follow the instructions to the Add a Printer Screen. Click Custom> Custom FFF Printer then Next. Now you need the specifications of your printer. Add in the print volume in mm into the Printer settings and the print bed size, and click Origin at Center and heated bed.



3 Checking the hotend

Click the second tab along the extruder and select the nozzle size. This should be listed in your printer specs, or if you have changed the nozzle, check the diameter and enter this here. Now add the Compatible material diameter of 1.75mm or 2.85mm and click Next. You're set to go!

GIMP

Credit: www.gimp.org

How to make better cut-outs in GIMP

We often have photos where certain elements have to be "isolated" from the background. Use GIMP and **Karsten Günther** for this task!

OUR EXPERT

Karsten Günther is a long-time LaTeX, Linux and GIMP user, and has written a number of books on these topics.

QUICK TIP

Don't change the preset threshold of the colour selection directly with the slider. Instead, use the interactive options, which change when you hold down the left mouse button. Move the mouse to the right and the threshold becomes larger; move it to the left and it becomes smaller.

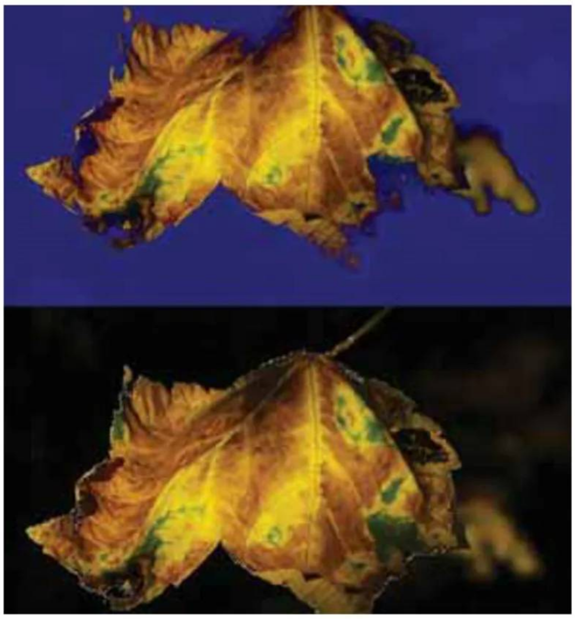
The isolated parts of an image contained on a separate layer can usually be used either in the same image, or another image as additional elements. They can also emphasise certain parts of the image. Every form of collage, for example, is based on this technique. We're going to look at a number of common methods that *GIMP* and the *G'MIC* plug-in offer for isolating image elements, often known as cutouts.

In principle, the task is quite simple. The desired parts of the image – the motif or the "foreground" – are marked with a selection and then either copied and pasted as a new layer. Alternatively, the selection is inverted and the rest is removed from this layer. Creating a clean selection isn't usually straightforward, which is why several selection tools and various methods of combining selections should be explored.

The first approach the user will probably try is to make use of just one of *GIMP* or *G'MIC's* selection tools. The result will differ greatly, depending on the subject to be extracted and its surroundings. The "magnetic scissors" tool is tricky to master, but foreground selection, for example, often delivers the goods.

In many cases, a multi-stage procedure makes sense. Starting with the rectangle selection tool means you create a "pre-selection" that contains everything that's needed while keeping the image size to a minimum. Smaller image sizes help to reduce your computing load, and processing time. Select the desired area and then paste it into the target image. Now you can convert it into a regular layer on which you'll now continue to work.

Which selection tool you use next to create the final selection essentially depends on the image and the motif to be extracted. The colour selection, the foreground selection and the free selection tools work well. Selections don't have to be created in one go, nor with just one tool. It's often much better to combine several selection tools. You can add to an existing selection while holding down Shift, and then remove parts of an existing selection by creating the new selection while holding down Ctrl.



The foreground selection tool is already quite good, but not perfect. Several minor adjustments are almost always necessary at the edges.

Foreground selection

The foreground selection tool has been revised several times in recent years and extended by various options. The tool works largely automatically, but requires a large amount of computing time. Two separate steps are involved when creating the selection. First, the approximate area around the motif is marked, then the motif is marked with a brush. Next, the option "Show preview" in the dialog that appears activates the calculation of a preview.

The developers recommend working first with the "Matting global" engine and then with the "Matting Levin" option. The problem with the more complex "Matting Levin" algorithm soon becomes apparent when the brush covers larger areas. The algorithm requires a disproportionate amount of resources and quickly paralyses even high-spec computers designed for image processing for a longer period of time. This doesn't occur with the "Matting global" algorithm, but the results are generally less precise. Even switching to the "Matting Levin" algorithm incurs a delay, which is necessary due to the recalculation involved.

If the foreground selection inadvertently detects parts belonging to the background as a motif, these are marked after choosing the corresponding option and thus removed from the selection. Usually some adjustments are necessary with this tool to achieve standout results. Whether it's worth putting in the time and (computing) effort depends largely on the images to be processed.

In the current version of the foreground selection, the following should be noted. First, in the dialog the option "Preview mask" must be active for the tool to start working. And second, sometimes it still doesn't seem to work. In this case, first check whether the option "Draw mode" is actually activated for what you want to select: foreground, background or unknown. Behind this are border areas in which the algorithm attempts to determine once again more precisely which parts belong to the foreground.

The brush isn't a real painting tool in this mode, but only simulates its behaviour, somewhat inaccurately. Each release of the mouse button starts a new calculation. Straight strokes (click, move, shift-click) aren't possible.

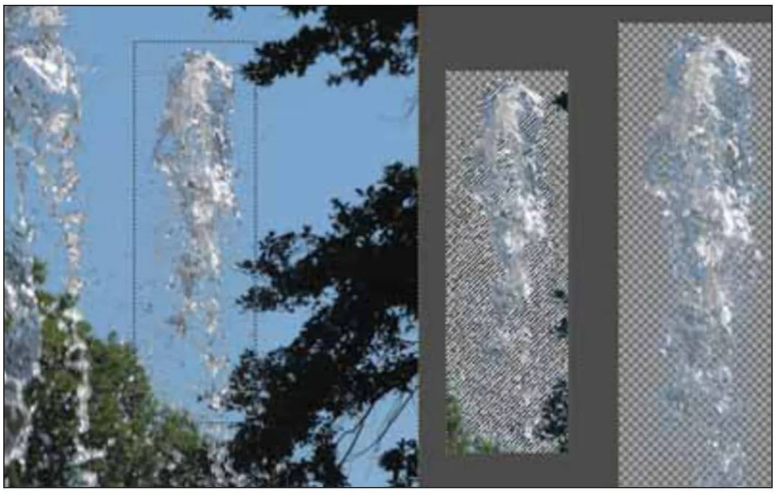
Inverse selections

In many situations, the colour selection tools help to create at least a good preliminary selection. In particular the fuzzy selection option succeeds in filling in small gaps in an existing selection. So the combination of these tools is a useful method for creating good selections. For the "repair" of small blemishes, the free selection is often a quick means of choice.

The example with the water fountain (*above right*) shows another problem. In this picture it's all but impossible to select the water itself. Therefore, in such cases the procedure is reversed. First, the desired area is copied quite closely with the rectangle selection on to a new layer. Only this layer is used for further work. The global colour selection ("Select by color") removes the sky, which is translucent in many places, and finally the Eraser tool removes the other parts in the edges. This example took about one minute to complete.

In practice, however, there comes a point in almost every image-editing task where the editor has to decide whether certain parts belong in the selection or not. This is completely independent of the tools used and occurs in almost all natural images. It's simply not possible to tell whether a small structure belongs to the foreground or the background. Then the editor decides what's appropriate in this situation.

Often this leads more experienced users to work only with the free selection, because they use it to rework the fine details. Then all decisions are made directly by the user and with some practice the procedure is often just as quick as using other selection tools.



The column of water from a fountain can be easily removed using inverse selection.

Free selection is one of *GIMP's* best and most effective tools. It's quick and easy to use, and really only has two disadvantages. Once the selection is closed, the supporting points can no longer be moved. Of course, the selection can still be improved. In addition, the free selection only handles straight or freehand drawn lines – the tool doesn't recognise curves.

Using paths

This is where the Path tool comes in. It's not a conventional selection tool as such, but creates (directed) lines in general. Provided these lines are closed, they can be converted directly into selections. There are two key features here. First, with paths the supporting points are retained. They can be subsequently moved, new ones added and existing ones deleted. Second, the line segments between the support points are curves and are thus well suited to working with rounded structures. This means that far fewer support points and segments are needed than would be the case with free selection.

However, users who have never worked with paths before often find the Path tool complicated and confusing to use. And yet the application is quite simple. When you first choose the tool, it's in Design mode. Each mouse click sets a new supporting point, which is initially connected with straight segments. The support points can be moved and the segments can be bent with the mouse.

What happens now is typical for paths. "Handles" are created at the support points with which the segments

QUICK TIP

Being able to display additional selected areas with a mask is helpful when finding appropriate threshold values.

» SELECTING IN PRACTICE

Remember that you're not limited to a single tool when creating a selection. Perfect selections are rarely achieved in practice with natural images anyway. Most of the time, using a combination of different methods is much more sensible and effective approach.

For example, Extract Foreground creates only a moderately good selection, which then fills a new transparent layer. Now some corrections can be made. Using "Select from Alpha Channel", this selection is transferred to the original layer and can be further improved with another tool – often this will be the local colour selection option. This selection can be edited again or used as a basis for a new selection.

Don't forget the Quickmask tool. This is activated via the button in the lower-left corner of the screen. Once it's activated you can use any painting tool in the foreground colour to paint over areas, which are then selected. Using the background colour you mark areas that should not be selected. Intermediate colours mark partially selected areas.

»

QUICK TIP

From the Selection editor, selections can be converted into paths. As always, this is done along the 50 per cent line.

can be brought into an almost arbitrary curve shape. To create a curved line, only a few straight segments are required, which can then be adapted exactly to the desired shape..

The tool can be used very effectively. First, control points are set around the motif, as best as possible, at the edges. In the example photograph there were about 50 at the beginning. By the end about 30 remained, and so probably even less would have been sufficient.

Next, the segments that were initially straight are adjusted individually with the mouse and bent so that they fit the contours better. *GIMP* then shows a small "handle" at the two end points of each segment, with which the curvature of the segment can be changed later. If you look closely, you'll see that there are actually two levers, each acting on one of the segments ending at the control point.

Often this handle isn't required, because the mouse already brings the necessary momentum into the line. It can then be ignored. The effect of the handle changes when you make it longer or shorter. This technique of working with Bézier curves is quite intuitive and easy to learn with just a bit of practice.

One advantage of this two-part method is that virtually all the fine work is combined in the second step, which involves adjusting the contours and reducing the number of control points. The key advantage of paths over freehand selections is that there are still exactly the same possibilities for improvement later on as there were at the beginning. Paths can be saved, reactivated, improved and combined to suit the image you're working on.

Objects that have been cropped can easily be inserted into other images on their own layer.

» EDIT MARGINS

The edges of the selected areas deserve special attention when using selections. The classic selection tools all have an option to create uniformly soft edges. Whether this is always the best option, however, depends on the circumstances. Sometimes it's better to work with a hard selection first and then rework it with a pre-configured eraser. To do this, use the option "Paint along the selection outline" in the selection editor (right button). Alternatively, hard selections can be softened at the edges using the Feather function, accessible from the selection menu.

In principle, the procedure can also be used for masks. After all, these are simple greyscale images. In this case, both processing methods are possible. Painting with black increases the size of the transparency area, while painting with white shifts the edge inwards and reduces transparency. Whether pens or brushes are used and which brush tips are used depends on the exact circumstances.

You may find yourself accidentally deleting individual segments. This isn't a problem, even if you don't notice it immediately and so are unable to use *GIMP's* Undo function. Click one of the endpoints and then, while holding down Ctrl, click the other endpoint. The two interlocking rings that appear mean that the path is now "married" here – in other words, closed. Note also that points accidentally set in this mode can be removed at any time using Ctrl+Shift+mouse click. The same applies to segments.

The Eraser tool

The advantage of the erasing method is that different variations can be tried out. Unsatisfactory results are simply undone using Undo (Ctrl+Z) and a new attempt is started with adjusted parameters. If the results are predominantly good, critical areas can of course be reworked manually. As always, the following applies here: erasing can be undone by holding down the Alt key, provided the edited layer has an alpha channel.

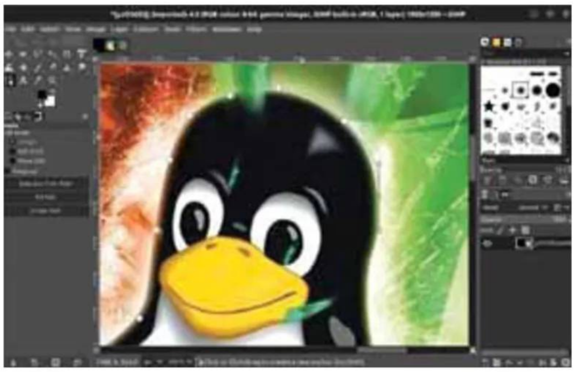
You can also use the Eraser as the main tool for cropping. On a duplicated layer, everything that's not required is simply erased. In order to make this visible, the (underlying) original layer should be invisible or at most with greatly reduced visibility. In fact, this is quite a practical method for otherwise difficult-to-work-with subjects, again if the layer has an alpha channel.

Layer masks

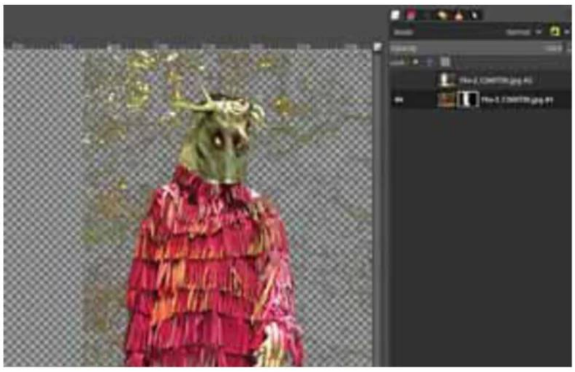
Making use of Layer masks is almost the same approach as the previous method of using the Eraser tool. Each layer, independent of the existence of an alpha channel, can be provided with an editable mask. The quickest way to do this is to use the context menu in the Layer dock. Masks only contain brightness information that determines the visibility of the content of the associate layer. In other words, they're greyscale images. Areas of white in the mask are displayed unchanged, grey areas appear with reducing opacity while black areas won't appear at all.

In the Layer dock, *GIMP* displays layers with masks with two preview images. The one on the right symbolises the mask and is selected with a mouse click before editing. Editing masks is an old technique that's independent from all other selection techniques.

When using masks to extract motifs from an image, masks from high-contrast layer variants are often



Paths can be easily edited to follow the edge of an object, be adjusting the anchor points along their length.



On the layer mask the brightness controls the visibility of pixels of the layer. Painting tools thus control them non-destructively.

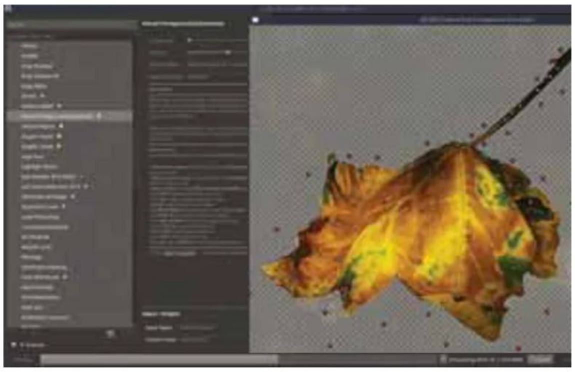
required. A duplicate of the original layer is processed for this – often with the Threshold tool – and the result is used as a layer mask. The same procedure was followed in the illustration layer mask (*above*). Any remaining gaps and superfluous material are retouched by painting tools using white and black.

With masks it's therefore possible to show desired parts of the image with a painting tool and to hide others. The fact that masks leave the original layer unchanged is an advantage that shouldn't be underestimated. Another great strength of masks is that if you provide a mask with a colour gradient, it's possible in a very simple way to let the layer "run out" and merge into another.

Enter G'MIC

G'MIC also offers a special treat for cropping. Since 2014, its chief developer David Tschumperlé has been developing an easy-to-use tool for cropping with the Extract Foreground (Interactive) filter. It works interactively in a special preview window that *G'MIC* opens automatically. With the left and right mouse buttons, you set control points in the preview window that mark areas of the image that either belong to the object or not. After setting some control points, pressing Space creates an updated representation in the preview window.

The tool has a number of special features. For example, it's not necessary to set a negative point using the right mouse button opposite each positive point. In most cases, setting the positive points is sufficient, which saves data and speeds up processing. Actually, negative points only have to be used if *G'MIC* fails to recognise areas of the background as part of the subject. Often just moving points is enough to improve the results.



Not yet perfect, but already quite good…G'MIC creates a layer with the extracted motif.

The Tab key can be used to switch the display several times, which is often useful to more precisely see what's currently selected. Pressing M enables you to do the same with the control points. Pressing Return ends the preview and processes the picture in full resolution.

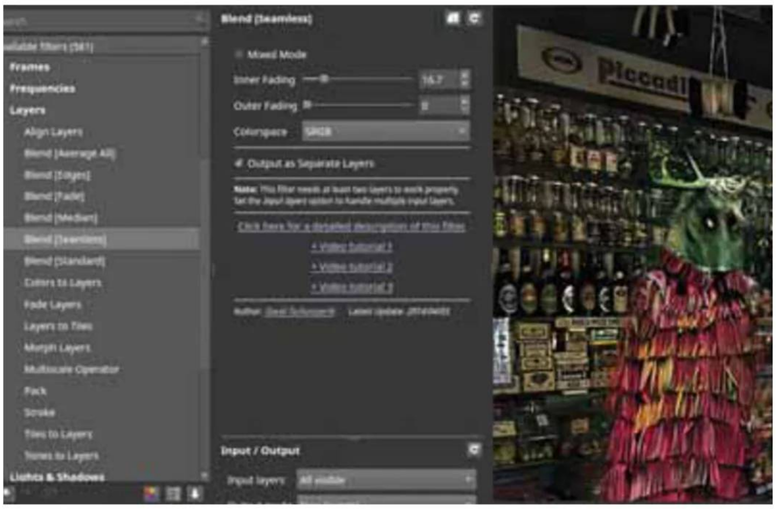
Once control points have been set, they can be moved with the mouse at any time and their effect changed (positive points can become negative and vice versa). This is even possible when editing has been completed with the Return key. *G'MIC* "remembers" the positions of the control points between calls to the programme. Other possible actions are explained in the dialog.

A disadvantage of this otherwise useful tool is the fact that only contiguous objects can be selected. With the water column, unfortunately, it fails all along the line. On the other hand, it's quick and easy to use, even for inexperienced users. However, note that the clippings created with this tool always have hard edges.

Seamless cloning

Of course, these are not all the methods available for extracting objects. An alternative, which is not so much about the exact cloning of the edges, should be briefly mentioned here: seamless cloning. This method involves cutting objects out with a clearly visible border. This border is used to create a fusion with the background. *GIMP* has had a tool for this for several years, but so far it's not totally stable.

G'MIC provides the same filter under the name Blend [Seamless] in the Layers section. To try it out, you should first use an image with just two layers: the background is the bottom layer, while above it the motif is inserted with a border on a separate layer. The rest is easy: activate the filter and test with the two sliders what the best options are. The illustration (*below*) shows this filter in action. LXF



The possibility of blending almost without visible transitions is promised by the Blend [Seamless] filter that's available from within G'MIC.

» IMPROVE YOUR LINUX SKILLS Subscribe now at http://bit.ly/LinuxFormat

STEGANOGRAPHY

Credit: www.libreoffice.org

The art of hiding messages in plain sight

Steganography might not be as well-known as cryptography but, as **Mike Bedford** reveals, it's a powerful tool in preserving your security.

OUR EXPERT

Mike Bedford used invisible ink as a kid. Many years later he learned it was called steganography and discovered it's gone digital.

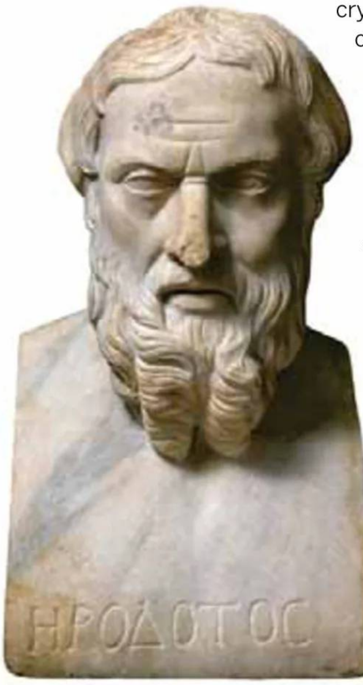
QUICK TIP

Most steganography is network steganography. However, the term has a specific meaning of embedding information in network control protocols. For example, information could be hidden in optional fields in a TCP/IP packet header.

The year is 499BC and Histiaeus tires of his stay in Susa. He conspires to secretly start a revolt in Ionia against the Persians, something that would give him the excuse for his return. But how to send a message to Aristagoras without discovery?

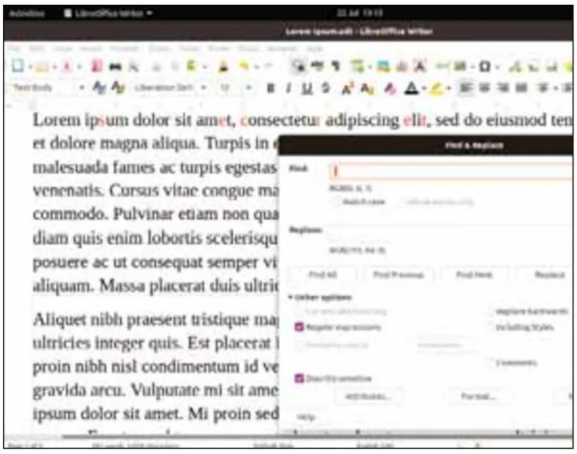
According to Herodotus, in his classic work *Histories* – which chronicled the Greco-Persian Wars, and is considered to be the first ever piece of historical literature – Histiaeus shaved his servant's head and tattooed that message on his scalp. Then, when the servant's hair had grown back, the servant was dispatched to Aristagoras who was instructed to shave the visitor's head to reveal what had been hidden. Yet if the servant had been apprehended en route, his captors wouldn't have realised that he was carrying sensitive information. While ultimately Histiaeus' ruse might have been realised and he was ultimately beheaded, this episode is thought to be the first-ever example of the technique of steganography – the word coming from the Greek for concealed writing.

Although the aim is the same, it's not the same as cryptography. In cryptography, no efforts are made to hide the presence of a message, so security comes from that message being sufficiently scrambled to make it unintelligible. In steganography, the message isn't transformed – although for additional security it can be encrypted first – but it's hidden in such a way that nobody will suspect that information is being exchanged.



According to Herodotus, there's a record of writing on to the wooden base of a wax tablet before applying its bees' wax writing surface.

CREDIT: Metropolitan Museum of Art, CC0 1.0, **https://commons.wikimedia.org/wiki/File:Cropped-removebg-herodotus-historian.png**



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod ten
et dolore magna aliqua. Turpis in
malesuada fames ac turpis egestas
venenatis. Cursus vitae congue ma
commodo. Pulvinar etiam non qua
diam quis enim lobortis scelerisqu
posuere ac ut consequat semper vi
aliquam. Massa placerat duis ultric

Aliquet nibh praesent tristique ma
ultricies integer quis. Est placerat
proin nibh nisl condimentum id ve
gravida arcu. Vulputate mi sit ame
ipsum dolor sit amet. Mi proin sed

Tiny alterations to the characters' colours in this LibreOffice Writer document were unnoticeable to the eye, but a simple find and replace has revealed the word "secret".

Here we're going to show you how to exchange messages using steganography, but don't worry, we won't be asking you to shave your head. Instead, we'll be hiding information in commonly exchanged digital media including images, audio files and even text documents. First, we'll provide some guidance on how to do that manually, so you get to see how it works, and then we'll follow that up by introducing software that makes steganography a more practical proposition.

Steganography in LibreOffice

To start, and to show you the sorts of techniques that are used in steganography, we're going to investigate a couple of ways in which data can be hidden in a *LibreOffice Writer* document.

First, select a document that you're going to use to act as the carrier for your clandestine message. Chose a fairly conventional document, by which we mean one that has a single colour of text, although it doesn't matter if there are coloured headings. The document must contain all the letters of your secret message, in the order they appear in that message. In practice, that probably means that the document will be quite a bit longer than your hidden message.

Open that document in *LibreOffice Writer*. We're assuming that the text is black, although the same principles apply if the text is a different colour. Now, starting at the beginning, find the first occurrence of the first letter in the message you want to hide. Select that letter and edit its colour from an RGB value of (0, 0, 0), which is black, to (0, 0, 1). It won't look any different, after all, it only differs from pure black by one part in 16.7 million, but this is exactly what we want since the whole purpose of the exercise is to hide information. Continue in this way until you've hidden each of the letters in your message.

Now, let's assume that you're the recipient of a document that's been doctored in this way, and which you know contains a hidden message. You could look at the colour of each letter in turn, writing down all the letters that have an RGB value of (0, 0, 1) to reveal the message. But that would be a long-winded and tedious process, so let's use a better way.

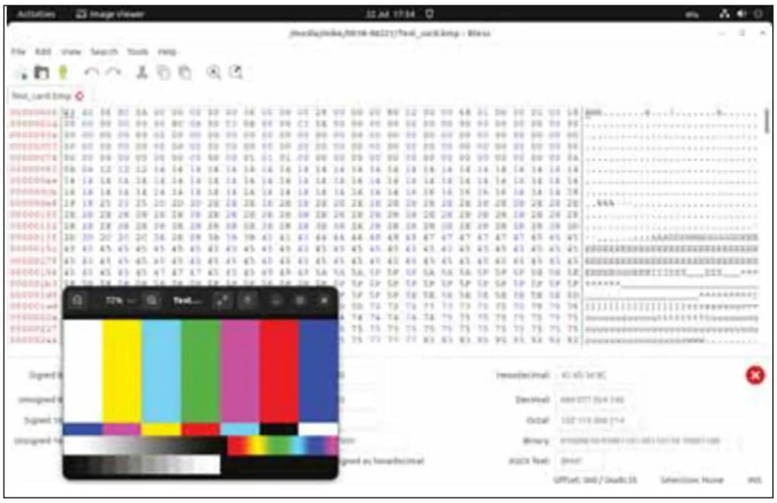
Undoubtedly you've used the search and replace feature in word processors to change all occurrences of one word or phrase to another. *LibreOffice Word's* search and replace facility is quite sophisticated and, although we'll leave you to figure out how to do it, it enables you to search for any letters that are a particular colour and replace them with the same letters in a different colour. So, for example, you could change all letters in our very slightly modified black to something noticeable, like red, so the concealed message stands out. Alternatively, you could change all letters in true back to white so all that remains visible is the formerly hidden message.

There are probably countless other methods you could conjure up to conceal a message in a textual document using *LibreOffice Writer*. However, we're going to look one particular method, because it sets the scene for our next topic of hiding a message in an image. And unlike our previous method, it enables you to embed any sort of data, not just text.

Fun with colour

The technique is similar to our first foray into steganography in that it involves manipulating the least significant bit of the colours of letters in the document. However, rather than that off-black colour being used just to draw attention to the associated letter, the hidden data is concealed in those least significant bits. Or, to put it another way, the least significant bits of the colour values of a block of letters represent binary data. It could be ASCII values of characters, or it could be something entirely different.

And now that we're dealing with binary data, it would be helpful if the process used to reveal that data presented it as a sequence of 0s and 1s so it can readily be exported to other applications. Easy – use *LibreOffice Writer's* search and replace, as before, but this time, instead of changing colours, change any character with an RGB value of (0, 0, 0) to a zero, and any character with an RGB value of (0, 0, 1) to a one.



Using a hex editor such as Bless, as used here to edit a BMP image of a test card, is a good way of learning how image steganography software works.

From text to pictures

Textual documents offer modest scope for camouflaging information because they tend to be of quite modest proportions. In particular, our method of hiding binary data in a *LibreOffice Writer* file had a maximum capacity of one bit per character. Bitmapped graphics files, by which we're talking about JPEGs, PNGs, TIFFs and the like, are much larger and less restrictive in this way, so they're commonly used as carriers in steganography.

The process becomes quite a bit more involved with formats like JPEG, which use lossy compression, so we'll

» WORKING WITH INVISIBLE INK

We've looked at ways of hiding data in digital media, but there are also techniques that apply to the printed page. One such example is invisible ink, and you might have played with it as a kid.

If not, why don't you try writing something on a piece of paper using a small paintbrush and lemon juice as the ink. When it's dry, the writing will be largely invisible but, if you heat the paper under a grill, the hidden writing will turn brown.

It's not very secure, partly because sending someone a plain sheet of paper might seem suspicious, and partly because the paper tends to wrinkle when you write on it with lemon juice. But printing a photo over it will go some way to addressing those concerns. Even so, we suggest you only try it for fun, not if your fortune depends on it.

You can buy invisible ink for inkjet printers, which results in printed content that becomes visible under ultraviolet light. They're not intended for sending hidden data, but can be used, say, to confirm the validity of documents like entry tickets. If you want to try them out, see **www.sfxc.co.uk**. More secure, because photographing in infrared is more involved than just shining a UV torch at a document, are invisible ink cartridges that are only visible if imaged in the infrared. Many security-related business tasks use this method.

UV fluorescent invisible inks for inkjet printers aren't super-secure, but they do have applications and make an interesting diversion.

CREDIT: From SFXD website (**www.sfxc.co.uk**). Used with permission.

QUICK TIP

Many image files include information other than the pure image – GPS location, camera make and so on – as Exif data. It has been proposed that data could be hidden in the Exif data, although it's not considered a secure option.

limit ourselves to seeing how to embed data in a BMP file since it's one of the simplest formats. To cut a long story short, the file starts with a variable length header, and this is followed by the pixel data in which somewhere between one and 32 bits are used to define the colour of each pixel, depending on the colour depth.

If you're writing software to handle any BMP file, you'll need to take note of several pieces of information in the header, for example the number of bits per pixel. However, we're going to simplify things by restricting ourselves to a 24-bit image, which can represent 16.7 million colours. So really, the only bit of header information you'll need to take note of is the value you'll find at address 0A (hex) which is the offset, as a 4-byte value, of the start of the pixel data.

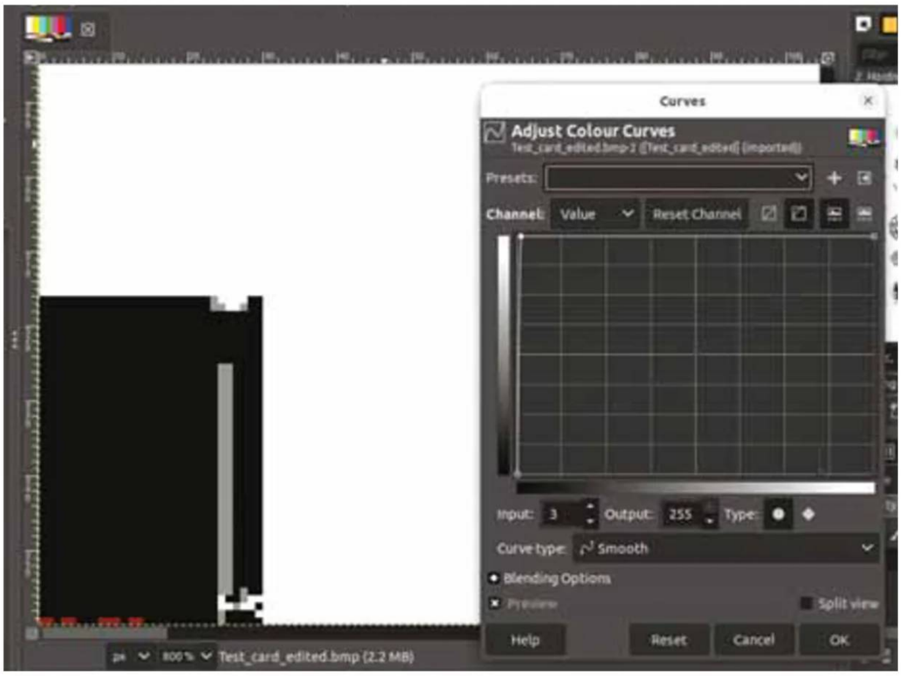
First of all, prepare yourself a BMP file to manipulate, so import an image into a photo editor such as *GIMP*, and export it as a BMP file, specifying 24 bits per pixel. Although it isn't a good choice for real-world steganography, as discussed in the No Solid Colours box (*overleaf*), if you use the test card image at **https://en.m.wikipedia.org/wiki/File:Test_card.png**, you'll see the same values as us. Open the BMP file in a hex editor. You could use one of the many command line editors, although we used the rather more sophisticated *Bless*. The online utility at **https://hexed.it** is another good choice for occasional use.

Now, make a note of the four bytes starting at 0A, which represents the address of the pixel data which you're going to modify. You'll find that those four bytes are 36, 00, 00, 00 and, because BMP files use the little endian approach, this means that the pixel data starts at address 00000036. So, look at the data starting at 36 and you'll see that there's a string of zeros which represents black. That data relates to the bottom line of the image because BMP's origin is the bottom-left hand corner, and you'll notice that it is, indeed, black.

As a first exercise, to give you confidence that the BMP data genuinely is as described, edit a few of the three-byte blocks (remember that we're using a file that has a 24-bit colour depth) starting at 36 from (00, 00, 00) to (FF, FF, FF). That would be useless for steganography, of course, but ideal for this initial exercise, because they'll stand out like a sore thumb, which is exactly what we want So, save a copy of the edited file and view it on-screen. You should see some white dots in the black area at the bottom-left corner of the test card so you know you're on the right lines.

And now, for something closer to proper steganography. Starting again with the unedited BMP test card image, edit the least significant bit of a few three-byte blocks at the start of the pixel data. Change them from (00, 00, 00) to (00, 00, 01). In that particular example, we're changing the least significant bit of the red component of the pixels' colour, because the colour information is in the order blue, green, red in BMP files.

You could hide a byte, for example a single ASCII character, in eight pixels, leaving the pixel data unchanged at (00, 00, 00) if the bit of the ASCII code is a zero, or editing it to (00, 00, 01) if the bit is a one. This time, when you view the edited image, you won't notice those altered pixels that are carrying your secret information. However, read the No Solid



Channel: Value
Input: 3
Output: 255
Curve type: Smooth

We might only have altered the least significant bit of the red components of a few pixels, but their presence can be revealed using image editing.

» NO SOLID COLOURS

Although we used a test card to demonstrate how image data could be edited by hand to embed hidden information, that image would be a poor choice for steganography. That's because it's composed almost entirely of large blocks of single colours. And changes to the colour value of pixels – even if only in the least significant bits – are more easily detectable if all the surrounding pixels have an identical colour value.

If you doubt that – and we acknowledge that you won't notice the altered pixels by eye – open your edited version of the test card in a photo editor such as *GIMP* and zoom well in on the bottom left-hand corner, which is where we recommended you edit a few pixels. They'll all look black, but first appearances can be deceptive.

Select the Curves tool and edit the shape of the curve so the output (vertical axis) rises very rapidly from zero to 100 per cent in just the very first few values of the input (horizontal) axis. This stretches very similar values – such as ones that differ only in their least significant bit – to occupy the full range of values from black to white. And, as a result, the pixels for which you changed the least significant bit of the red component will truly appear as red.

Had we chosen an ordinary photo, and avoided embedding data in regions of fairly uniform colour, it would be nearly as easy to detect edited pixels.

Colours box (*below*) where we see how that data can be detected with nothing more sophisticated that a basic image editing tool.

An interesting thing to experiment with could be moving away from using the least significant bits of successive 24-bit pixel values, starting at the bottom left of the image. We can imagine that this would be the first place a would-be attacker might investigate, so surely there are better ways. Here we suggest just a few simple things to try but, if you get serious about this project, you'll want to learn about methods that enable the presence of hidden data to be revealed.

First, it makes sense not to use the least significant bit of consecutive pixels but, instead, to use some subset. The easiest way of doing that is for both parties to agree in advance about identity of the pixels containing the hidden data. Similarly, the data doesn't always have to be in the least significant bit of the last of the three bytes that represent a pixel. Instead of always using the least significant bit of the red component, the last significant bits of the green or blue component could be used instead, swapping between them in a pre-arranged sequence. And a final possibility is not to attempt to hide all of the data in a single image, but to distribute it between several images. In that case, you should avoid those images ever appearing together on a webpage, or being attached to the same email.

Getting practical

There's no shortage of free open source steganography software for Linux – both command line and GUI applications, some supporting only image carriers, some adding audio files, a few adding video carriers – and at first sight most seem to do the same thing except for differences in the supported file formats.

There are also some online utilities, although the tool we've chosen runs locally. It's called *Stegosuite* and you'll find it in the main repositories. It's a GUI application, and it's fairly intuitive to use. It doesn't support audio files, but for image files there's support for BMPs, GIFs, JPGs and PNGs, the latter being a drop-off with some tools.

Stegosuite's user interface couldn't be much simpler. When you first fire it up you're presented with a minimalistic screen indicating that no image has been loaded, and with just a couple of menus: File and Help. Assuming you want to start by embedding some data in an image, select your file at File>Open, and you'll see it'll be displayed in *Stegosuite*, and boxes will appear where you can provide more information.

Type your message

First of all, there's a box in which you can type a secret message and there's a box that shows a list of files that you want to hide in the image. It's not a case of just one or the other, though: you can choose to hide either a text message, a file (which can be anything), or both. If you want to add a data file, right click in the relevant box and select Add File.

Finally, enter a password in the relevant box – this will be the password that someone will need to use when they subsequently want to extract the hidden message and/or file(s) from the image – and click on Embed. Unless it encounters an error – for example if you're trying to embed too much data for the size of the image – a message will tell you that it's embedding data and shortly after it'll tell you that it's finished. You'll find the image with the hidden data in the same folder as your original image, and its filename will be the same except that it will have "-embed" appended to the name. If you're going to send that file to someone else you should rename it, because the presence of the "-embed" might draw unwanted attention, should it fall into the wrong hands. After all, the whole point of steganography is that nobody will suspect that you're exchanging any information other than the blatantly obvious.

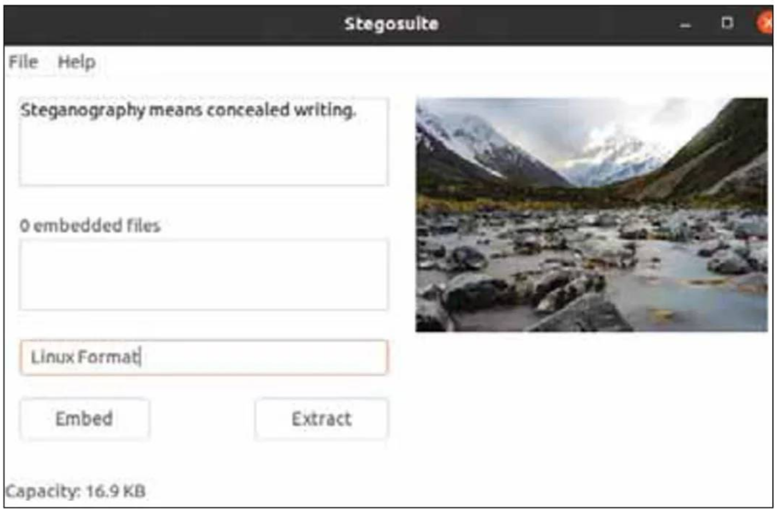
Extracting hidden data from a file is just as straightforward. Open the image, just as you did for embedding data, make sure there's nothing in the message or files boxes, enter the password, and click Extract. So long as the password is correct, and the image did indeed contain hidden information which had been embedded by *Stegosuite* (because other steganography software won't necessarily use the same technique), the extracted information will be presented to you. In particular, if there was a text message it'll appear in the message box, and if there were embedded files, their filename(s) will be shown in the files box.

There's one other thing we should tell you, which applies to *Stegosuite* or any other image steganography tool. It's advised that you don't use a carrier image that's available online. If someone intercepts the file into which you'd hidden some data, then if the base image is available online it's an easy task to find and download it. A file compare would then reveal if the two files are identical and, if they're not, suspicions may be raised.

We chose to feature *Stegosuite* because we wanted to introduce you to a practical tool for basic steganography without having to spend forever explaining how to use it. However, if you're serious about your privacy then you could do better. In closing, therefore, we'll mention *OpenPuff*, which is much more fully-featured than *Stegosuite* and appears to be well respected. LXF

QUICK TIP

A watermark can be added to images to protect the copyright holder and it can be either visible or invisible. Invisible digital watermarking is similar, in some ways, to steganography, therefore, even though its purpose is different. Unique to watermarking, though, is that the method aims to make the watermark recoverable even if the image has need cropped or otherwise edited.



Steganography means concealed writing.

0 embedded files

Linux Format

Capacity: 16.9 KB

It might be simple in the extreme, but Stegosuite provides the basics for embedding secret data and, and extracting it from, an image file.

» **FIND MORE HIDDEN MISTAKES…** Subscribe now at **http://bit.ly/LinuxFormat**

NOURISHMENT FOR THE SOUL

Andrew Lukoshko, benny Vasquez and Jack Aboutboul update **Jonni Bidwell** on the latest AlmaLinux developments.

Back in 2014 Red Hat partnered with CentOS, which for years had been rebuilding and repackaging the sources for Red Hat Enterprise Linux (RHEL) and distributing them under a free licence. Some pundits were curiously optimistic by this odd alliance. A RedMonk writer wrote, "This move by Red Hat represents the logical embrace of an adjacent ecosystem." Most were puzzled and some were deeply cynical. Erstwhile *Linux Format* regular columnist Doc Brown, never one to mince his words, suggested replacing "embrace" by "stranglehold" in the previous quote, and suggested that "adjacent ecosystem" was "Dilbert-speak for competitor". Brown's words were borne out when Red Hat announced in 2020 that it would be sunsetting CentOS in favour of its own CentOS Stream, which would track ahead, not behind, RHEL.

In response to this, a number of RHEL downstreams were established. One was Rocky Linux, set up by CentOS co-founder Greg Kurtzer. Another was AlmaLinux, which aims to be a community-owned and governed, forever-free enterprise Linux distribution. We've covered AlmaLinux before (see **LXF282**), but much has happened since then. It's got a new release, a new build system and a new project (ELevate) to enable migrations between major versions of the various enterprise Linuxes. And AlmaLinux OS foundation chair benny Vasquez, architect Andrew Lukoshko and community manager Jack Aboutboul were kind enough to tell us all about them. Read on to find out how the AlmaLinux project continues to bring soul to the enterprise Linux sphere.

AlmaLinux 9.0 was released at the end of May 2022, just a few days after its upstream, Red Hat Enterprise Linux 9.0, hit the mirrors. That's an impressive achievement for a project whose first stable release took place around a year ago.

You may have missed our previous coverage or be unfamiliar with AlmaLinux, so here's what the project is about in the words of AlmaLinux OS foundation chair benny Vasquez (lowercase intentional): "AlmaLinux OS is an open-source Linux operating system that stepped into the gap left by the discontinuation of the CentOS Linux stable release. AlmaLinux is community-owned and governed, forever-free and focused on long-term stability. The distribution provides a robust production-grade platform and is 1:1 binary compatible with RHEL and pre-Stream CentOS Linux.

"The distribution is called AlmaLinux because 'Alma' means 'soul' in Spanish, and 'nourishing' in Latin; two things that we believe are critical to providing AlmaLinux. Quite simply, it's a homage to the best parts of what the entire Linux ecosystem has achieved. Having a soul is critical to the future of open source everywhere. Open source software development isn't the same as other ways of making a product; it relies on people wanting to collaborate to be successful. That's hard to manufacture and hard work to do right, but it's also incredibly rewarding."

Learning from the past

We study history in order not to repeat the errors of the past. And with hindsight it's apparent that CentOS was marred by issues of leadership and governance. So how is AlmaLinux going to avoid these problems?

As benny explains, community and not commodification is key: "When it came to the governance model for the project, we looked at several ways to structure things. One thing that became glaringly clear was that we had to consider the history and develop a structure that avoided the things that brought the CentOS users to where we were. Without getting into all those details, CentOS seemed to be treated more as a 'product' and changed hands several times, the last of which was the acquisition by Red Hat. So we took expert advice and carefully considered the best way forward for AlmaLinux. The goal was to create the most transparency and put the ownership of the OS, the Intellectual Property, and the project's direction into the community's hands."

AlmaLinux
POWERFUL.
FREE.
ON PREM.
IN THE CLOUD
AS A CONTAIN
The community owned and govern CentOS Alternativ
ALMALINUX.ORG
AlmaLinux
almalinux.org

The AlmaLinux stand at the Open Source Strategy Forum with Ampere.

Empowering the community thusly seems like a recipe for success, a strong foundation, if you will: "There were various non-profit and for-profit options available to us, but we decided to make the AlmaLinux OS Foundation a 501(c)(6) non-profit as that ensured that the entire community controls the assets related to AlmaLinux OS, which prevents them from being suddenly sold in the future. Originally, the foundation was set up by Igor Seletskiy, CEO of CloudLinux, and initially we used resources from CloudLinux to bootstrap the project. But there's ongoing transparency work, for instance, with assets and the build environment, which the foundation has undertaken to ensure that we are independent and Open Source.

"To avoid any confusion over the community ownership of the project and to foster transparency, Igor stepped down from the board last year. In 2021, I became chair of AlmaLinux OS Foundation, and while we have board members from various sponsors, we'll be conducting an election soon to foster greater diversity and expand the board."

Lately, the AlmaLinux team has been working on a new build system. Project architect Andrew Lukoshko gives us some background: "We built the first releases of

» DEEP POCKETS AND A DELUGE OF DOWNLOADS

The concept of AlmaLinux sounds edifying and eminently sustainable. But what about finances? Who is providing the proverbial Benjamins for AlmaLinux? Well, there are a fair few sources, as benny explains. "AlmaLinux OS Foundation members, sponsors and partners back the AlmaLinux OS with investments and long support commitments. AlmaLinux OS enjoys at least $1M in annual sponsorship from CloudLinux Inc and support from many other sponsors, including Algolia, AMD, ARM, AWS, Black Host, CloudFest, Codenotary, cPanel, Equinix, Hivelocity, Knownhost, Orcharhino, Procomputers.com, Progress Chef, Mattermost, Microsoft Azure, Rackspeed and World4You. Ongoing development efforts are governed by the members of the community."

Well, we can't argue with an alphabetised list, and it's important to note that big names are using it as well as supporting it. "AlmaLinux has been adopted and is supported by some of today's foremost Open Source ecosystem projects including VMWare, GitLab, MongoDB, Tenable, and more," says benny. And they're using it in big numbers, too. "The distribution has millions of downloads from a network of over 200 mirrors worldwide and over 1.5 million Docker pulls, and availability on all major public cloud platforms including AWS, Azure, Alibaba, Google Cloud, Oracle OCI and others."

AlmaLinux Foundation Chair benny Vasquez enjoys crocheting, dev-ops and upsetting trenchant views on first-name capitalisation

»

AlmaLinux in the CloudLinux infrastructure, in an internal build system. But we understood from the very beginning that AlmaLinux would need a new public build system that would enable us to be completely open and allow users to see how we build and test our packages. AlmaLinux 8.5 for PowerPC was the first release built entirely in the new build system.

"During the preparation of this release, we found and fixed bugs, made various improvements, and soon the system was ready to bootstrap the AlmaLinux 9.0 Beta for all architectures. Finally, the production readiness of the build system was confirmed by using it for the release of stable versions of AlmaLinux 8.6 and 9.0 in two and eight days after the corresponding RHEL release, respectively."

So the AlmaLinux Build System (ALBS) is now building AlmaLinux. This does seem (semantically) like the way it should be. But ALBS has been in the works for some time and, as Andrew explains, lessons from CloudLinux have informed its development.

"The history of the AlmaLinux Build System began with lessons learned while developing a build system for CloudLinux's internal needs which, when it was developed a decade ago, existing build systems could not cover all the needs of the company. AlmaLinux genuinely has very similar needs, which are to streamline a pipeline for building source code from various sources; support for signing the kernel, modules and bootloader for Secure Boot; deep integration with a code review system; advanced tools for testing built packages; a convenient release process into internal or public repositories to build beta or stable; and even the ability to roll out updates in small steps to avoid the massive problems caused by a possible unstable update. All the accumulated experience over a decade was applied to the development of the ALBS – there's no greater teacher than experience."

AlmaLinux OS architect Andrew Lukoshko helped elevate the ELevate project.

Sine Nomine's Neale Ferguson was key in helping AlmaLinux conquer the mainframe.

Besides ALBS the AlmaLinux team has been working on another ground-breaking project, this time called ELevate. Community manager Jack Aboutboul (whom readers may remember from our previous Alma content) and Andrew both describe ELevate as "a project and a set of tools to help enable major version migrations between versions in the enterprise Linux ecosystem, hence the name. It's meant to work across the ecosystem, so this is a distribution-agnostic tool. This was a conscious effort to help bring the community together to work jointly on the project."

A Leapp of faith

ELevate is based on a technology called Leapp, a Python framework for managing RHEL upgrades. You'll find the documentation for Leapp at **https://leapp.readthedocs.io/en/latest** and if you're anything like us you'll be baffled by it. We posited to Andrew that taking this complex tool and making something useful across distros must've been a considerable amount of work.

"It was a lot of work, but it was certainly work that we were glad to do," he says. "We simplified migration for users by preparing ready-to-use packages for upgrading to any popular EL-derived distribution, not only AlmaLinux. Also, the original Leapp only supports RHEL, so some modifications to the code was required to support other operating systems. But in general, this is

» WORKING WITH LIVE MEDIA

Linux Format may not have a cover-mounted DVD anymore (sparing Jonni hours of *xorriso*-related headaches), but someone still has to master the ISO images for Alma's live media. AlmaLinux committer and community member Bala Raman explains that its live media continues to evolve.

"Currently, x86_64 and aarch64 have support for UEFI. GRUB support is available in x86_64, aarch64 and ppc64le. When creating a custom DVD ISO, adding `grub2-efi-*64*` to the package list will enable UEFI support, and `package shim-*64` will give a Secure Boot option. Adding the `isomd5sum` package will provide the menu option to test/verify the custom ISO created. If anyone is looking to get a deeper understanding of DVD boot layout, studying the manifest files under **https://repo.almalinux.org/almalinux/9.0** will give more insights.

"To put together Live images, we need a list of packages, an image-builder tool (*livecd-tools* or *lorax*), and a live-image script. Essentially, it's like cooking a nice meal with multiple ingredients and following a recipe.

"Upstream RHEL 8.6 and RHEL 9 are now using Wayland as the primary display server (which is a modern, secure, and more straightforward windowing system) and moving away from X11. GNOME 40 provides full support for Wayland. With the help of Red Hat and other community developers, KDE and Xfce are becoming stable in EPEL9, and more work is in progress for EPEL8 while KDE provides initial Wayland support now."

Community man Bala Raman told us that the AlmaLinux team misses the LXF DVD.

still the same Leapp, and we never intended to fork it and develop it independently; we need more cooperation, not less. We're in touch with the Leapp maintainers and talk to them frequently. We're working on providing pull requests, and the maintainers will decide which of them to include in the Leapp source code.

"In terms of human resources, it required me as a project leader, one developer for the Package Evolution Service, and another developer to help with patches for Leapp. You don't need a lot of people when you have an experienced team and a great community working together!"

So with a few modest modifications to Leapp, ELevate was ready for action. Almost. As Andrew explains there's always a sting in the tail: "Leapp requires several configuration files to run, the main ones being a .repo file with the repositories of the target operating system, as well as a huge JSON file called **pes-events.json** that describes the changes that happen to packages between major releases.

"According to the Leapp concept, a package can be present, removed, deprecated, replaced, split, merged, moved or renamed. Without this information, it's impossible to carry out a correct system upgrade as DNF update simply will not work. While all the other configuration files are prepared by hand and put into the ELevate project git, this JSON file required the development of the Package Evolution Service (**https://pes.almalinux.org**), enabling you to add and edit entries, as well as dump data to the configuration file for Leapp. The service also allows users or organisations to create their own datasets."

Looking around the slightly intimidating Leapp documentation in Red Hat's knowledge base, we found some references to an already existing Leapp dataset. Rich with data though this file is, Andrew and Jack are quick to point out it's not much help to anyone but RHEL subscribers: "Red Hat provides its own dataset for Leapp, but it has two problems. First, it's not freely available and requires a specific Red Hat subscription; its licence also does not allow us to use or distribute it.

"Second, this dataset only supports RHEL, and we would still have to modify it to support other operating systems. We actually had a great collaboration with Oracle here. Oracle was working on an Apache-licenced metadata set, and so we worked together on integrating that. We are super-grateful to them for this and believe that without this help, the project would not have been possible. The Oracle dataset was used as the initial set and then enhanced to support other EL-derived distributions using the Package Evolution Service."

Community manager Jack Aboutboul has helped connect AlmaLinux with resources, expertise and high-energy physics.

Applications
AlmaLinux

Thanks to Bala's efforts, there's now an XFCE Spin of the AlmaLinux ISO available.

RHEL made a name for itself partly by being able to run on HPC and mainframe (particular IBM's Z) architecture from early in the game. Now AlmaLinux can run on Z, as well as 64-bit PowerPC (PPC64le). As Jack says, sometimes it's best to bring in the experts: "One of our goals for AlmaLinux is to be an exemplary Open Source citizen and help continue to promote and facilitate collaboration and cooperation among the various groups in the community. For alternative architectures, it's always important to get support from experts in those architectures and work together with them to make sure that you have a holistic effort around dev, testing, refining and so on to make sure things are buttoned up. If you look at IBM Z, a group called Sine Nomine (SNA) had already been building what they called ClefOS from the CentOS sources. Some community members connected our dev teams, and we started discussing an effort around building EL 8, and we could see that this would be a great collaboration."

Neale Ferguson, chief scientist at SNA, explains that they also had help, this time from the academic sphere: "The resources for the AlmaLinux on z build were provided by Marist College as part of its Linux Foundation involvement. Marist also provided the infrastructure used to build ClefOS before it reached end of life.

"It has to be said that none of SNA's Linux on IBM Z distributions would have been possible without Marist College's resources. Our first builds of ClefOS 6 were in a farm of Intel servers running Linux on an emulated mainframe. Originally, the turnaround time was in the order of six weeks to get a release built from scratch. We now have 10 virtual machines that can pump out a release in seven to 10 days and we hope to be able to scale up the AlmaLinux infrastructure as releases age out."

Jack tells us that it was a similar story for the ppc64le architecture: "A similar spirit of collaboration to AlmaLinux on Z held true for ppc64le. Oregon State's Open Source Lab was instrumental in helping us with hardware resources and technical knowledge around the architecture. Lance Albertson and his crew are true gems in the community. We found that CERN [once home to some 40,000 CentOS installs] researchers also happened to have PowerPC machines around. We've been in touch with them for a while and they started using the images right away, validating the work we had done and putting it to good use unlocking the secrets of the universe!" **LXF**

techradar.
The home of technology
techradar.com

THE BEST NEW OPEN SOURCE SOFTWARE ON THE PLANET

HotPicks

Syncthing » Imagine » HomeBank » Jitsi » Beets » vifm » Rclone » Unvanquished » Bitfighter » AES Crypt » SMPlayer

Mayank Sharma is at the stage in his life where he doesn't mind admitting that he gets a chuckle every time he mistypes HotPicks as HotPics.

FILE SYNC

Syncthing

Version: 1.20.4
Web: https://syncthing.net

Taking a backup is often suggested as the best means to guard against accidental data loss. While that's a prudent strategy, another could be to use a file-sync tool such as *Syncthing*, which can easily duplicate your sensitive data on another computer on the network, or across the internet.

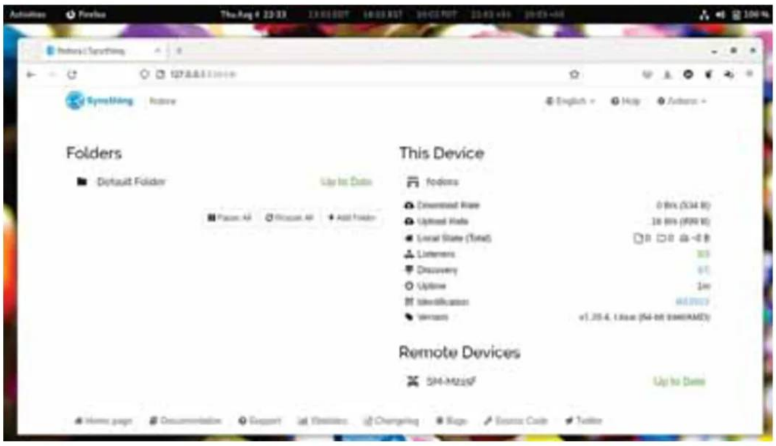
There's no need to install the *Syncthing* server – you can just download, extract and run it. When you run *Syncthing* from a terminal as a normal user, it'll fire up a browser and take you to its administration interface. By default, this interface is only accessible from the same computer, but head to Actions>Settings>GUI and change the GUI listen address to **0.0.0.0:8384** to access it from any computer. Remember, however, to also set a GUI username and password to lock access to the administration interface.

For *Syncthing* to be able to sync files with another device, it must be paired with that device by exchanging device IDs (or scan its QR code via the *Syncthing* app) that are unique cryptographically secure identifiers. By default it'll automatically sync the contents of the default holder (**~/Sync**), and you can add more directories and select the computer you wish to share them with, using the Add Folder button.

During sync, all data is encrypted using TLS and transferred using direct peer-to-peer connections. This means files go directly from folder to folder, without touching any centralised cloud servers.

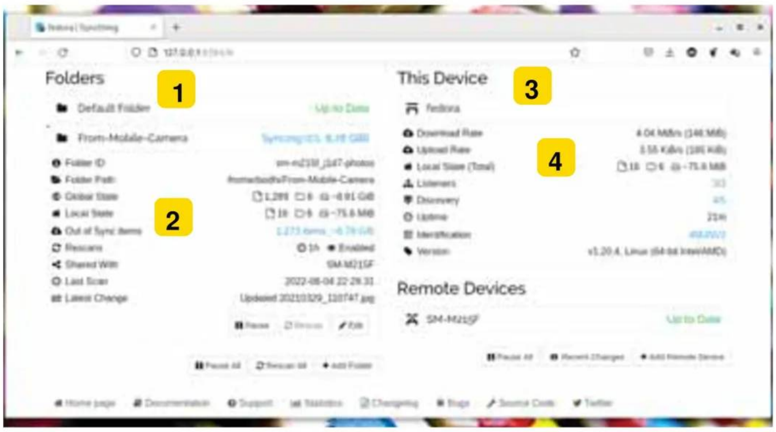
While sharing, you can instruct *Syncthing* to not change files on the original folder when they're changed on a remote computer. If you enable this, only changes made to the files in the original computer will be sent to the remote shares, but not the other way round.

You can use *Syncthing* to share a folder with specified computers, and choose from one of its multiple versioning strategies. These and other useful syncing functions are explained in its detailed user manual. Make sure you spend some time flipping through it, despite the programs's intuitive interface.



Syncthing has one of the best application dashboards that we've seen – it's tremendously verbose yet not overwhelming.

GET TO KNOW SYNCTHING'S DASHBOARD



1 **Folder view**
The left-hand side of the screen shows the ID and current state of all configured folders. A folder can be in several status such as Unshared, Syncing and more.

2 **View details**
Clicking the folder name makes that section expand to show more detailed folder information, and buttons for pausing the folder, forcing a rescan or editing the configuration.

3 **Device view**
The right-hand side of the screen shows the overall state of all configured devices. The local device is always at the top, with remote devices in alphabetical order below.

4 **Device details**
For each device you see its current state and, when expanded, more detailed information such as its transfer rates and the amount of data transferred.

IMAGE OPTIMISER

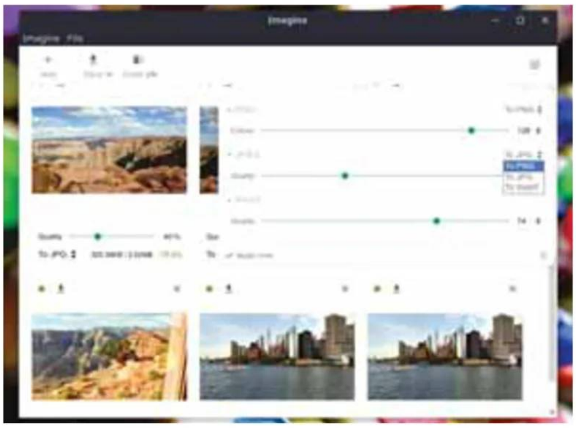
Imagine

Version: 0.6.1
Web: https://github.com/meowtec

While reducing image size isn't all that difficult, to do so without compromising on image quality can be challenging. Thankfully, this is an area where *Imagine* excels.

You won't find *Imagine* in software repositories of most modern desktop distros. But that's fine because the project is distributed as an AppImage package. Download the latest .AppImage from the project's GitHub Releases page, and run `chmod +x Imagine-*.AppImage` to make the file executable. Launch *Imagine* with the `./Imagine-*.AppImage` command, or by double-clicking the file from a file manager.

The tool features an intuitive interface, and you can click the Add button at the top-left to add files, or drag and drop them into the interface. All images are displayed as thumbnails, with pertinent information, such as file size, displayed at the bottom. You can use the quality adjustment slider to adjust the compression rate. The tool will show the change in file size as well as a percentage as you adjust the value on the slider. You can also use the drop-down menu at the bottom-left of the thumbnail if you wish to convert the file format. The tool supports JPG, PNG and WebP image formats.



Imagine is a handy tool for someone who regularly processes images, especially for posting online.

You can use the Settings button on the top-right of the interface to define global settings. It will enable you to define the colour of PNG files and quality of JPG files. You can also define the output format for PNG and JPG files. For instance, if you set the quality as 75, and output as PNG for JPG files, the tool will automatically perform these operations when you add any JPG files.

Click an image after *Imagine* has completed the operation to take advantage of the tool's Before/After preview engine, which provides a clear view of the impact the image optimisation process has on your pictures' quality. When you're finished, you can either save and process the images individually, or batch-save them with a new name.

PERSONAL ACCOUNTING

HomeBank

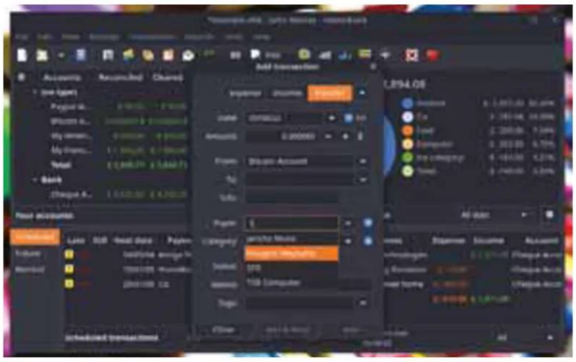
Version: 5.5.6
Web: http://homebank.free.fr/en

Ever notice how many sci-fi movies often find a way to remind us that maths is constant across the universe? Closer to home, worrying about income and expenses is just as much a constant, with most people struggling to keep tabs on their spending during the current cost of living crisis.

HomeBank is a personal finance and money management tool to help you track the flow of funds. It even boasts of powerful filtering tools and graphs, which will enable you to analyse your spending habits and identify where savings can be made.

You'll find *HomeBank* in the software repositories of most popular desktop distros. The project doesn't provide an easy-to-install binary, meaning you have to run the old trusted trifecta of `./configure`, `make` and `make install` if you wish to manually install it on your system. Ubuntu users can optionally take advantage of the PPA on offer.

You'll be greeted with a welcome screen when you first launch *HomeBank*. When you create a new file, the File Setup wizard will guide you through the necessary steps. *HomeBank* will also create many useful categories such as Alimony, Car Payment, Gasoline, Bank Charges, Bills and such, with additional sub-categories under each. For instance, the Bills category has separate sections for cable, electricity, cell phone and more.



HomeBank has an intimidating interface that'll only start making sense once you've gone through its user manual.

The project supports a wide number of currencies. While *HomeBank* supports importing bank statements, it only accepts ORF, QRF or QIF file formats. If you must use CSV then you'll have to format the file to be acceptable for *HomeBank*, as described in the documentation. In addition to bank accounts, you can also use the application to manage your credit card spending, as well as track cash, and other assets and liabilities.

It might sound too cumbersome to begin with, but once you get used to it, you'll be able to configure *HomeBank* to alert you about potential overdraft on an account, and even anticipate future costs.

VIDEO CONFERENCING

Jitsi

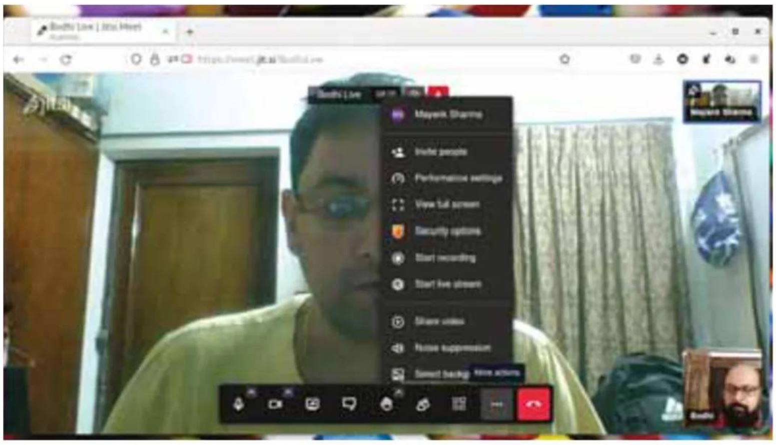
Version: 2.10
Web: https://desktop.jitsi.org

The *Jitsi* project isn't a single program, but rather a set of applications. The project's namesake application is a VoIP and instant messaging tool that has an impressive set of audio and video conferencing features.

As a VoIP client, *Jitsi* enables you to make calls using the Session Initiation Protocol (SIP). It has the usual softphone features in that it can mute, put on hold, transfer and record calls. It can also make registrar-less SIP calls to other *Jitsi* users on the local network.

You can use *Jitsi* to make video calls to one user or to several on both SIP and XMPP networks. *Jitsi* is also a very capable IM client with several unique features, including the ability to stream and share the desktop of users at both ends of the call at the same time.

The desktop version of *Jitsi* is no longer actively maintained by the *Jitsi* team, but by the community. They put out pre-compiled binaries for the client for both Deb and RPM-based distros. For installation you can download the RPM files and install it on top of Fedora with `sudo dnf localinstall jitsi*.rpm`.



Since anybody with the meeting room can join a conference, Jitsi Meet offers security options to restrict access.

The *Jitsi* project is now best known for its *Jitsi Meet* video conferencing platform. You can use *Jitsi Meet* to host video conferences via the browser on your desktop or through its smartphone app. Simply head to **https://meet.jit.si** and choose a name for your virtual meeting, or perhaps even accept one of the amusing ones the website generates randomly.

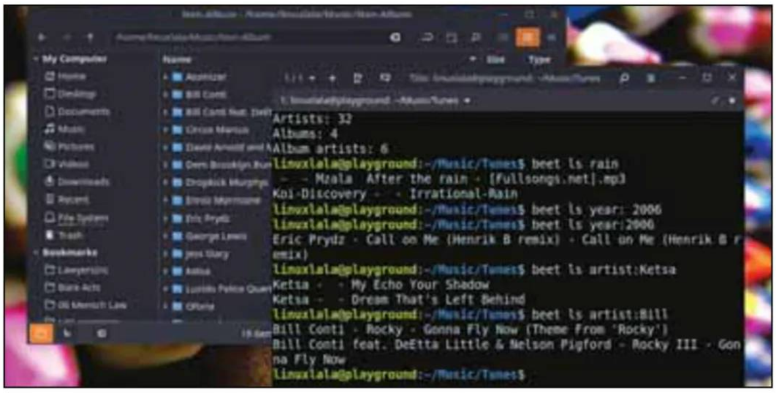
Once you've created your conference, you can share a link to it with your contacts from within the meeting room. They can also dial in and connect with a PIN.

Obviously *Jitsi's* free service has some limitations, but these can be overcome by self-hosting your own *Jitsi* installation. Refer to our Build a Smart Home Office feature in **LXF268** for step-by-step instructions on how to self-host a *Jitsi Meet* instance.

MUSIC LIBRARY MANAGER

Beets

Version: 1.6.0
Web: https://beets.io



```
Artists: 32
Albums: 4
Album artists: 6
linuxlala@playground:~/Music/Tunes$ beet ls rain
 - - Mzala  After the rain - [Fullsongs.net].mp3
Koi-Discovery - - Irrational-Rain
linuxlala@playground:~/Music/Tunes$ beet ls year: 2006
linuxlala@playground:~/Music/Tunes$ beet ls year:2006
Eric Prydz - Call on Me (Henrik B remix) - Call on Me (Henrik B r
emix)
linuxlala@playground:~/Music/Tunes$ beet ls artist:Ketsa
Ketsa - - My Echo Your Shadow
Ketsa - - Dream That's Left Behind
linuxlala@playground:~/Music/Tunes$ beet ls artist:Bill
Bill Conti - Rocky - Gonna Fly Now (Theme From 'Rocky')
Bill Conti feat. DeEtta Little & Nelson Pigford - Rocky III - Gon
na Fly Now
linuxlala@playground:~/Music/Tunes$
```

Beets supports the use of query strings with many of its commands, such as beet ls to help you quickly find the song you're looking for.

Back before the world switched to online streaming services (*speak for yourself youngster!–ED*) for their multimedia needs, music was collected and stored on disk drives. If this is a familiar scenario you might enjoy working with *Beets*.

The project strives to help you organise your music collection once and for all. Using the MusicBrainz database, the tool automatically updates the metadata on your music files as it catalogues your collection. This includes album art, lyrics, genres, tempo and much more. You can also fetch metadata from Discogs and Beatport, or even have *Beets* guess the metadata using the songs' filename or the acoustic fingerprint.

These and many other features are available courtesy of the myriad of plugins that can be used to further enhance the functionality of *Beets*. For instance, plugins can help identify duplicate tracks and albums, identify albums that are missing tracks, and also transcode audio to another format.

Most distros feature an older version of *Beets* in their software repos. The easiest way to install the latest version is with *Pip*, the package manager for Python projects. Ubuntu and Debian users can run `sudo apt install python3-pip` to install *Pip*, while those using RPM-based distros such as Fedora can do so with the `sudo dnf install python3` command, before running `pip3 install --user beets` to install *Beets*.

Once installed, launch *Beets* by pointing it to the directory housing your music collection with `beet imports <directory path>`. The tool will immediately report if it encounters any tracks for which it can't find a matching release. It will similarly inform you if it finds multiple matching candidates for a track, and ask you to choose the one that best fits your music file.

When done, you'll find *Beets* would have created a whole new directory structure to sort your files into meaningful directories, and also amended the file names, tags and other relevant details.

CLI FILE MANAGER

Vifm

Version: 0.12
Web: https://vifm.info

Your desktop environment already has a graphical file manager, but there also exists a variety of text-based file managers. *Vifm* is one such file manager, and its claim to fame is its adherence to a *Vi*-style key bindings. This has made it popular enough to be part of the software repos of many popular desktop distros.

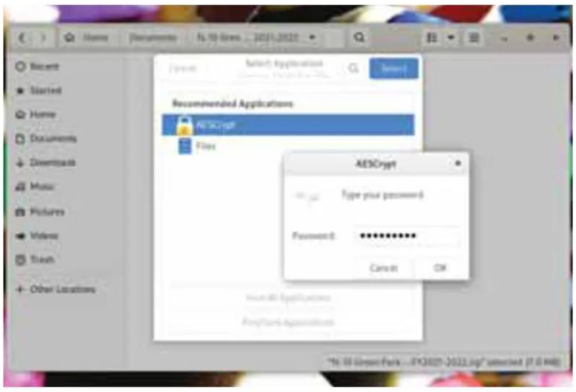
If your current distro has an older release on offer, you can download the AppImage binary from the project's website. Next, make the file executable by running the `chmod +x vifm-*.AppImage` command. You can then launch *Vifm* with the `./vifm-*.AppImage` command.

The project boasts of some very useful and convenient features, such as the ability to create bookmarks for frequently accessed directories. While this is common for many graphical and text-based file managers, *Vifm* stands out in that it also makes it possible for you to bookmark files.

Vifm defaults to a two-pane interface by default. You can move up and down a directory using the k and j keys. Press h or the left arrow key to move to the parent directory, and the right arrow key to navigate into a directory. The Tab key will switch between panes, while pressing l after selecting a file will open it in the associated application.

You can delete a selected file by using *dd*, before confirming the action with the y key. If you change your mind about any deleted file, note that they're relegated to a trash directory, which you can access using the `:trashes` command from within *Vifm*.

It's also possible to switch to a horizontal split interface by pressing Ctrl+w, followed by s. Using Ctrl+w followed by v will switch back to a vertical split. Press Ctrl+w for a list of all the keybindings for pane manipulation, such as increasing or decreasing its size, switching to a different pane, switching to a single pane interface and more.



Vifm would be intuitive to operate for someone used to working with Vi. Everyone else will fail to see the point.

CLOUD STORAGE SYNC

Rclone

Version: 1.59.0
Web: https://rclone.org

There's no such thing as being too cautious when it comes to protecting sensitive data. It's not just malicious attacks and external threats, either – something as innocuous as a misdirected `dd` command can wipe out all your data. Backups are the only way to ensure that your data remains safe from known, unknown and even imaginary threats.

Rclone is a nifty little CLI tool that supports over 40 different cloud storage solutions to help you safeguard your data on remote sites. You'll find it in the software repos of most popular desktop distros, but the project also offers a distro-neutral binary that you can place in a directory in your **$PATH** to complete the installation.

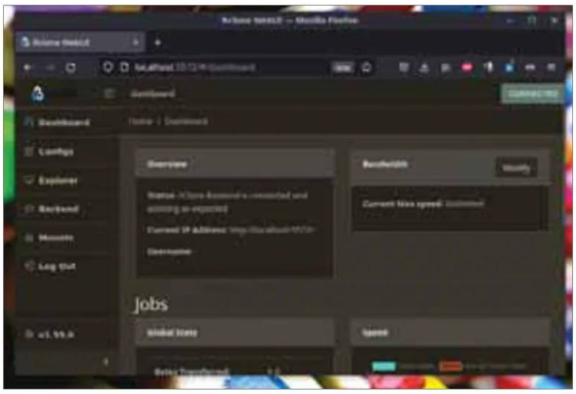
If that sounds like too much work, you can instead use the install script with `curl https://rclone.org/install.sh | sudo bash` to install the latest version.

With the installation done, run `rclone config` to get started. If this is your first time running *Rclone*, the project will use default settings to create a **~/.config/rclone/rclone.conf** file, and enable you to set up a new remote site for storing your backups.

The project will provide a numbered list of 49 cloud storage services and protocols including Amazon, Seafile, Microsoft OneDrive, Mega, Google Drive, Dropbox and FTP. The tool will guide you through the steps to help it connect to the cloud service, in order to facilitate the creation of encrypted backups on the cloud. Helpfully, it'll even point to its incredibly thorough documentation where needed.

Each such backup can be mounted as a disk, and you can also mirror the cloud data to another cloud service or a local disk, and even use tools such as *ncdu* (NCurses Disk Usage) and others to analyse the data stored in the cloud.

Should you ever tire of working from the comfort of the terminal, run the `rclone rcd –rc-web-gui` command to access the tool's browser-based graphical interface, although note this is still under development.



New users can experience the dexterity of rclone through the tool's web interface, which supports all of the commonly used functionality of the CLI tool.

FIRST-PERSON SHOOTER

Unvanquished

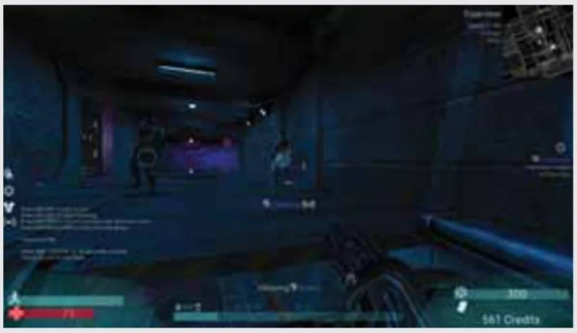
Version: 0.53 Beta
Web: https://unvanquished.net

On the face of it, *Unvanquished* appears like any other first-person shooter set in a futuristic, sci-fi setting. But the game has a rich history and was one of the first cool-looking open source shooters that could stand up to its proprietary brethren both in terms of gameplay and looks.

Unvanquished requires about 600MB of disk space, though of course playing online will involve downloading several extra content packs, such as mods and custom game levels. The game has modest hardware requirements and would work flawlessly even on a dual-core machine with 4GB of RAM.

Installing the cross-platform game on Linux is fairly straightforward. Just grab its compressed launcher, extract it and fire it up. The first time around it'll ask you for permission to grab the game, though on subsequent launches you can play the game straightaway, at least until there's an update.

Unvanquished combines a first-person shooter with elements of a real-time strategy game. It pits technologically advanced human soldiers against highly adaptable aliens. Players can choose to play from either camp, and will experience an entirely different adventure on both sides.



Based on the daemon gaming engine, Unvanquished is a completely open source fork of the popular Tremulous game.

The goal of the game is simple: prevent members of the opposite camp from spawning, and work together with your team to destroy the enemy's base. Along the way players earn upgrades based both on their individual performance as well as of their team. Upgrades for the humans come in the form of more powerful and effective weapons, while the aliens level up in the form of larger, nastier avatars.

The latest version of the game comes about a year after the previous one, and thus includes a plethora of changes, both to the backend and to the frontend of the game. For one, the developers claim to have improved the game's AI, which translates into more intelligent bots and team mates. There have also been several changes to the weaponry in order to improve gameplay.

MULTI-PLAYER ARCADE GAME

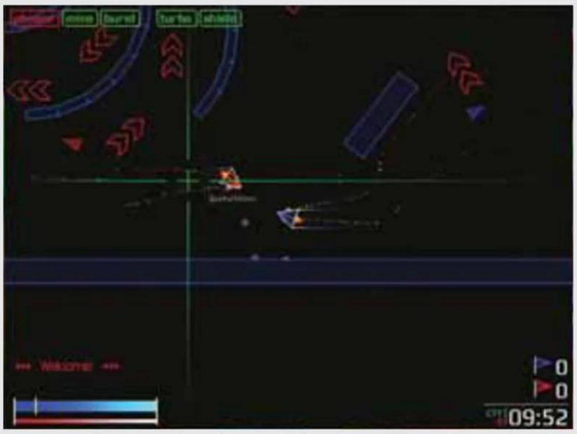
Bitfighter

Version: 022
Web: https://bitfighter.org

Bitfighter is a team-based multiplayer game that involves navigating your space ship across the vastness of space, as you collect flags, shoot enemy ships and evade incoming gun fire. This 2D arcade game requires dexterity of limbs and quickness that would surprise many FPS aficionados. You might even get a jolt from the how to play instructions, which span 17 pages!

You won't find *Bitfighter* in the software repos of many distros, but thankfully the project releases an AppImage that renders installation virtually moot. Download the AppImage from the project's website, and run `chmod +x bitfighter-*.AppImage` to make the file executable. You can now launch the game with `./bitfighter-*.AppImage`, or by double-clicking it from a file manager.

The first step is to decide if you wish to host your own game, or join an online community. Either way, we recommend reading the how to play instructions first, to master the controls. You control the ship's movement with the usual W, A, S, D keys, while using the mouse and the left-click to align the ship and fire at enemy crafts. The screen is riddled with updates about actions involving your teammates, and you can even chat with your mates to coordinate attacks and devise strategies.



Frenetic gameplay coupled with 80s-arcade graphics and a thumping soundtrack… strap yourself into Bitfighter today!

Your ship can be equipped with a choice of three weapons, and two modules. You can choose between options such as booster, shield, armour, repair, cloak and more, which you must use strategically to achieve your objective. Techno music plays in the background to keep you engaged, which begins as soon as you launch the game.

Don't complain to us if the fast-paced, heart-pumping retro music coupled with the quickness with which you're destroyed makes you fall off the chair or beat your keyboard in frustration!

FILE ENCRYPTER

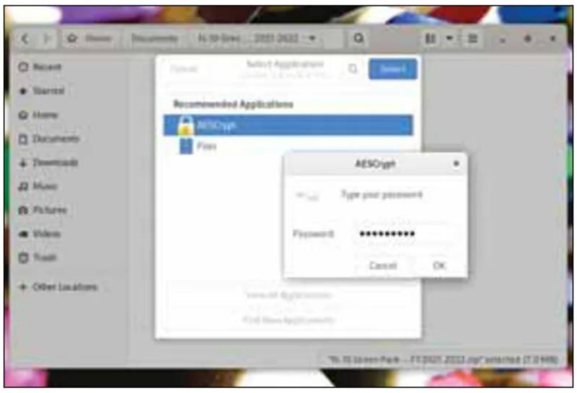
AES Crypt

Version: 3.11
Web: www.aescrypt.com

We don't have to convince you about the need to encrypt your files. Keeping files behind usernames and passwords no longer cuts it. But while there's no dearth of encryption tools, most of the good ones come with a learning curve. Furthermore, using them requires firing up another program to encrypt the files, and then navigating through menus to choose the appropriate settings. The usability break discourages most people from securing their files.

AES Crypt is a welcome break from all this jumping-through-hoops malarkey. It's one of the simplest encryption tools to install and use. Simply download and extract the installer from the tarball on its website. Then run the installer, which will prompt for superuser privileges before launching the graphical installer.

To use the tool to encrypt a file, simply right-click a file and select the relevant option in your file manager to open the file with another application. This will bring up a list of all the programs installed in your distribution. Select *AES Crypt*, which will prompt you for a password to encrypt the file twice. That's all there is to it. Your distro will help shave off a couple of seconds on subsequent use, since most file managers will now list the application in the right-click context-menu for quicker access.



AES Crypt also has a CLI version that offers a little more dexterity than the graphical version, including the use of a keyfile.

The encrypted version of the file will be placed in the same folder with the same name, but with an .aes extension. To decrypt the file you can open the .aes file with *AES Crypt* from the right-click context-menu. It'll prompt you for the password before decrypting the file in the same folder.

If you want to reap the benefits of encryption with the least amount of time investment, then there's no beating *AES Crypt*. However its simplicity comes at the cost of some flexibility, which might limit its appeal. The biggest limitation is that it only supports one encryption cipher (AES), which is a dead giveaway from its name.

MEDIA PLAYER

SMPlayer

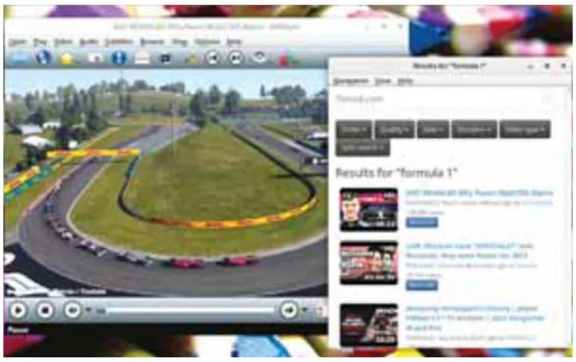
Version: 22.7.0
Web: https://smplayer.info

A cross-platform graphical frontend to the venerable *mplayer* command-line media player, *SMPlayer* can play virtually all known formats in circulation. The application is pretty similar to *VLC* in terms of form and function.

While the default interface lacks finesse and looks like an over-glossed cousin of *VLC*, you can change its look and feel by selecting one of the four alternative interfaces. Besides a minimal interface, there's also one that lets you drape it in a new skin. You can also switch to a custom icon set and widgets style.

SMPlayer offers several installation options. You'll find it in the official repos of every major desktop distro, and it's also available as a Flatpak, an AppImage and a Snap package. If you have snap support enabled in your distro, a simple `sudo snap install smplayer` will anchor it in your distro.

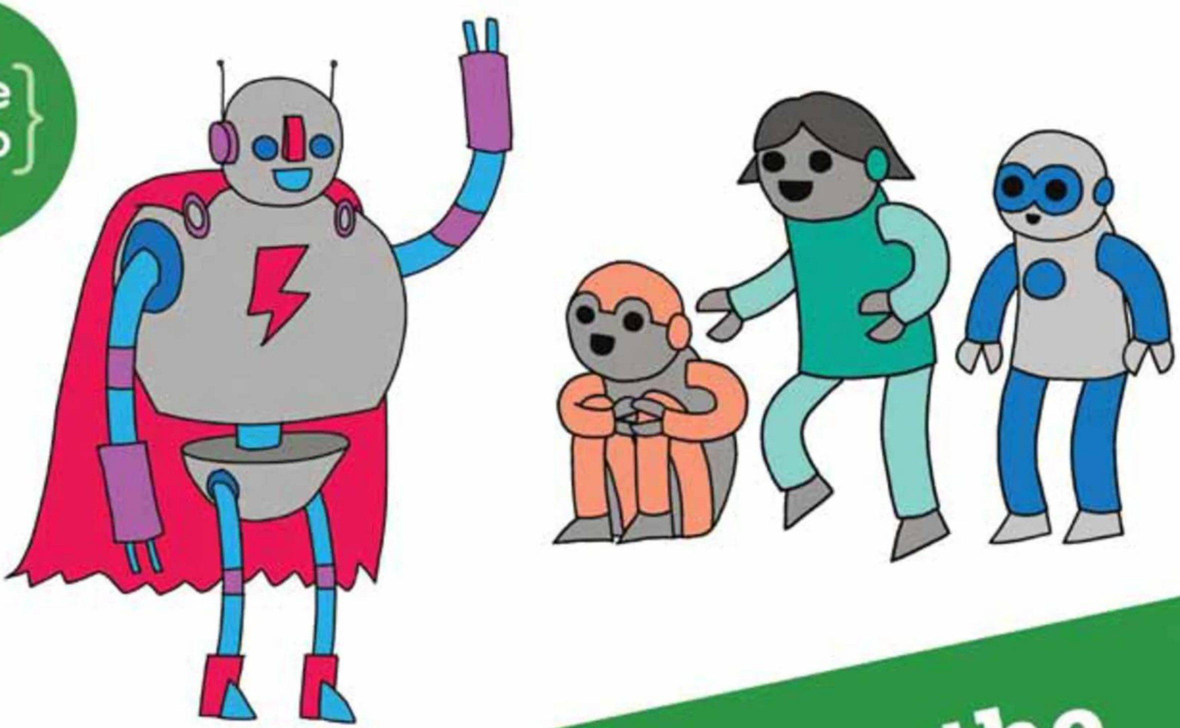
Just like *VLC*, *SMPlayer* also remembers the last position in a previously played file. However, unlike *VLC* that starts playback from the beginning but gives you the option to resume playback from where you left off, *SMPlayer* jumps to it automatically. Advanced users can also pass extra options to the *mplayer* backend.



SMPlayer also includes several backends that you can use to stream online content from YouTube and other websites.

Talking of backends, in addition to *mplayer*, the application also makes it possible for you to switch to *mpv*. *SMPlayer* also has a command-line interface and enables you to define keyboard shortcuts and custom actions for the mouse and the mouse wheel. It also has options for hardware acceleration on supported devices.

The highlight of *SMPlayer* are its configurable parameters, these might overwhelm first-time and inexperienced users. However, they'll appreciate the tooltips in the options window that help determine the purpose of the various parameters. Furthermore, *SMPlayer* has a good support structure with forums and FAQs, and there's a resourceful Help section within the program as well, which will help new users get the most from it. LXF



Can you help inspire the next generation of coders?

Code Club is a nationwide network of volunteer-led after school clubs for children aged 9-11.

We're always looking for people with coding skills to volunteer to run a club at their local primary school, library or community centre for an hour a week.

You can team up with colleagues, a teacher will be there to support you and we provide all the materials you'll need to help get children excited about digital making.

There are loads of ways to get involved!

So to find out more, join us at **www.codeclub.org.uk**

3jw8GGqdiRsHphOFjEgfZw==

CODING ACADEMY

PYTHON

Recreate the classic Pac-Man in Python

Being chased down endless corridors by spectral apparitions is all in a day's work for **Andrew Smith** and his Python skills...

OUR EXPERT

Andrew Smith is a software developer for NHS Digital, and likes video gaming and scripting in Python

This issue we're going to look at a game of *Pac-Man* created in Python using the PyGame library. The code for this project has been adapted from **https://pacmancode.com** where the original source code can be downloaded.

We'll install and setup Python 3.10 for the benefit of this tutorial. For those that have Python/PyGame already installed, Python 3.8+ should be a good-enough version to use with this tutorial. Type the following to install Python 3.10 and PyGame:

```
$ sudo apt-get update
$ sudo apt-get install python3.10
$ sudo apt-get install python3-pip
$ python3.10 -m pip install pygame
```

This project uses the numpy Python library and can be installed using *pip* by using the following commands if it's not already installed.

```
$ pip install numpy
```

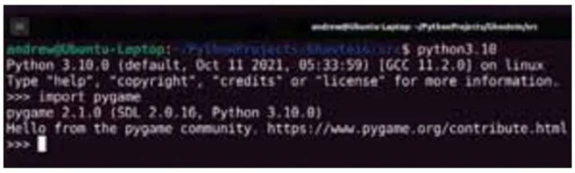
To ensure you're calling the correct *pip* version, check the **Scripts** folder of the Python installation folder. There may be a version number of *pip* to use, such as *pip3.8*. Check the versions of python and pygame, too. Next, `git clone` from repository:

```
$ git clone https://github.com/asmith1979/lxf294_
pacman/
```

The project has been placed in a folder called **PythonProjects**, which was created before downloading the project. The source code and project can also be retrieved from the *Linux Format* archive. This tutorial will focus on the source code located in the folder called **lxf294_pacman**. If you're not already in that folder type `cd lxf294_pacman` to access the Python source code.

To edit and view the source code you can either use a default text editor or you could use something more specific such as *Notepad++*, *PyCharm* or *VS Code*. In this tutorial, we'll be using *gedit* to view and edit the source files with. If and when using this method to view/edit source files, consider opening two console windows. Use one for editing/viewing source files and the other terminal window for executing the PyGame code

This game project consists of a lot of Python, image and text files that are required to run the project. Not all of the Python code will be covered in this tutorial, but a selection of some of the Python files will be studied, including **run.py**, **sprites.py**, **constants.py**, **pacman.py** and **ghosts.py**.



```
andrew@Ubuntu-Laptop:[illegible]$ python3.10
Python 3.10.0 (default, Oct 11 2021, 05:33:59) [GCC 11.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import pygame
pygame 2.1.0 (SDL 2.0.16, Python 3.10.0)
Hello from the pygame community. https://www.pygame.org/contribute.html
>>> 
```

Behold mortal a successful Python 3.10 and PyGame installation after going into the Python interpreter.

Running the project

Run.py is the starting script of the project that mainly consists of a class called `GameController`. This enables you to start the game, set up the background, initialise values and game states, update aspects of the game, render the graphics, and use various gameplay and helper methods.

Before we look at the code, have a go at running the *Pac-Man* project. Make sure you're in the folder with the source code and then type:

```
$ python run.py
```

When the game starts, it starts in a "Paused" mode. Press the Spacebar to start the game. When you do un-pause the game, the *Pac-Man* character will start to move around the game map and the ghosts will start to hunt for you.

You can use the cursor keys to navigate *Pac-Man* around the game map, When you want to quit the game, press the Escape key which would safely terminate the game program and bring you back to the command prompt of the terminal.

Near the bottom of **run.py**, you'll find code that's used to create an instance of the `GameController` class so that it can start the game. Let's look at the code that does this.

```
if __name__ == "__main__":
  game = GameController()
  game.startGame()
  while True:
    game.update()
```

QUICK TIP

Call your variables something meaningful so that you can easily identify what part they play in the program.

The main entry point to the program is specified by `if __name__=="__main__":`. After the `GameController` class is initialised, the main game loop for the program is started. This is a while loop, which can be seen from the above example. For this loop to terminate, a False condition must be given. The game is started by `game.startGame()`. The following process takes place after the `startGame` method is called:

- Data and Images for the maze map are loaded
- *Pac-man* game character initialised
- Ghost game characters initialised
- Configure ghost character behaviours

The update method (`game.update()`), which is called as part of the main game loop, is mainly used to continually monitor the game state, detecting for keyboard input from a player for example, or recognising collisions between *Pac-Man* and the ghost characters. It's also responsible for rendering the graphics for both the game characters and full game scene. While we're still looking at **run.py**, let's look at the class constructor of `GameController`:

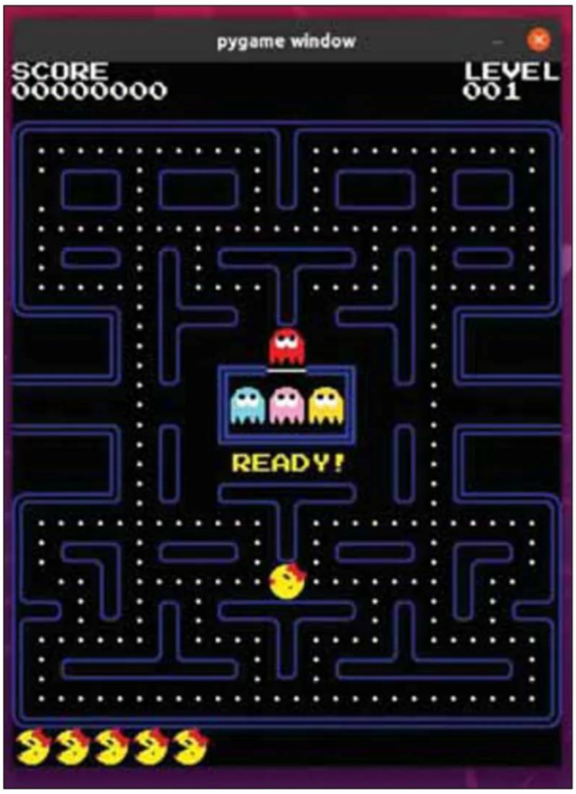
```
class GameController(object):
    def __init__(self):
        pygame.init()
        self.screen = pygame.display.set_
mode(SCREENSIZE, 0, 32)
        self.background = None
        self.background_norm = None
        self.background_flash = None
        self.clock = pygame.time.Clock()
        self.fruit = None
        self.pause = Pause(True)
        self.level = 0
        self.lives = 5
        self.score = 0
        self.textgroup = TextGroup()
```

Shown above is part of the constructor of the `GameController` class. Like any other constructor, it's used to initialise various game values and states. You may see from studying the code in the constructor that you can alter the level you start at, how many lives you start with as well as the score and many other aspects. Before continuing, have a go at changing the level you are on or the lives that you start with. Then run the program again to see the effect of the changes that you've made. When you've finished playing the game with the changes you made and are ready to continue, press the Escape key to exit the program.

Pause and quit the game

Another part of the `GameController` class that's worth looking at is how the pause and quit functionality to the game program have been added. In your chosen editor for **run.py**, find a method called `checkEvents` and have a look at the code below:

```
elif event.type == KEYDOWN:
        if event.key == K_SPACE:
            if self.pacman.alive:
                self.pause.setPause(playerPaused=True)
                if not self.pause.paused:
                    self.textgroup.hideText()
                    self.showEntities()
                else:
                    self.textgroup.showText(PAUSETXT)
                    #self.hideEntities()
        if event.key == K_ESCAPE:
            # exit()
            self.escapeKeyPressed = True
```



A new game of Pac-Man is about to start. It is just us, or are all the ghosts doing a collective eye roll?

From part of the `checkEvents` code shown above, you'll see that to detect a keyboard press the `KEYDOWN` event type is used. `K_SPACE` is the key for the spacebar and `K_ESCAPE` is the Escape key.

Now that we've covered how the project is started, we'll now look at how all the images are loaded for the game project. Find and open a Python script file called **sprites.py** in your editor. You'll see that **sprites.py** consists of the following classes:

- SpriteSheet (Base class)

CLASSES FOR CLYDE

Object Orientated Programming (OOP) is a programming paradigm based on the concepts of objects, classes, properties and methods. It includes techniques such as inheritance and polymorphism. OOP has been used a lot throughout this project and can be considered to be a project that is fully object orientated because all the operations relating to the game and characters within it are implemented in classes that are closely related with each other.

A good example to study are the classes implemented in **ghosts.py**. The first class that's declared is `Ghost`: this contains properties and methods that all ghost characters will use during gameplay. From viewing **ghosts.py**, you'll find that the concept of Inheritance is used a lot in the implementation of these classes. For example, the `Ghost` class is used as the based class, and other implemented classes such as `Blinky`, `Inky`, `Pinky` and `Clyde` all inherit the properties and methods of the `Ghost` class. Another example where the concepts of inheritance is used a lot is in the Python script file, **sprites.py**. In this file there's a base class called `SpriteSheet` and the other classes in the file such as `PacmanSprites` and `GhostSprites` all inherit the properties and methods of the `SpriteSheet` class.

QUICK TIP

Add comments to your code to help explain how segments of code work more. This may prove useful if you revisit a project several months after starting it.

- PacmanSprites
- GhostSprites
- FruitSprites
- LifeSprites
- MazeSprites

The first class shown in **sprites.py** is `SpriteSheet` and is known as the base class because the other classes that are declared in this script file inherit the properties and functionality from this class. The `SpriteSheet` class consists of a constructor that's used to load in a sprite sheet image file (**spritesheet_mspacman.png**) and has a method called `getImage` that's used to grab an image from the sprite sheet loaded in with the constructor.

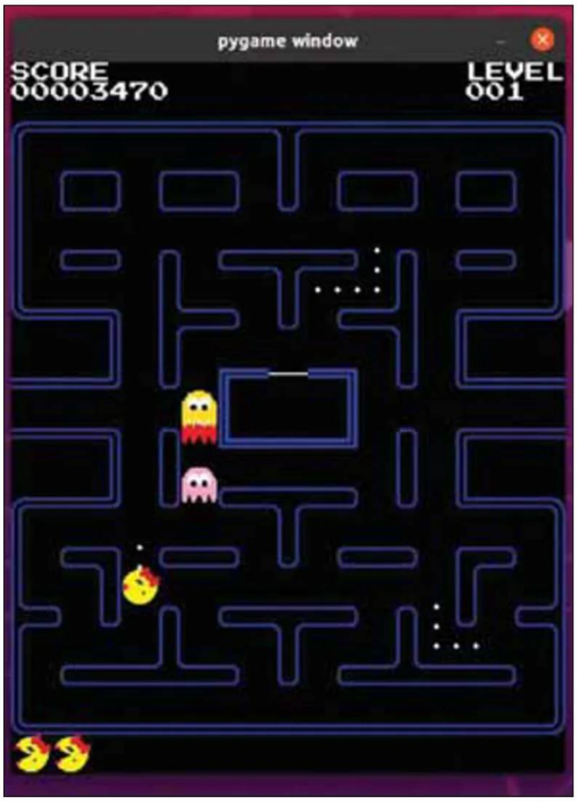
The class following on from this, `PacmanSprites`, is used to load all the images from the sprite sheet that contain *Pac-Man* images.

Open up the image file **spritesheet_mspacman.jpg**. You'll see that there are many different images from each aspect of the game: *Pac-Man* character images, ghost images, maze images and so on. The images for *Pac-Man* and the ghost characters are collated together to then perform an animation while the game program is executing. You'll see from viewing the sprite sheet image that there are eight separate images to animate the motion of *Pac-Man*. As already explained, this class inherits its properties and functionality from the `SpriteSheet` class. To see how this is done, view the code example below:

```
class PacmanSprites(Spritesheet):
  def __init__(self,entity):
    Spritesheet.__init__(self)
```

To include the properties and functionality of `SpriteSheet`, the class name is put in as an argument when declaring the `PacmanSprites` class. Furthermore, in the constructor of `PacmanSprites`, there's a call to the constructor of the `SpriteSheet` class to initialise values in the base class first and then the derived class `PacmanSprites`. The same is done in all of the remaining classes: `GhostSprites`, `FruitSprites`, `LifeSprites` and `MazeSprites`.

The `getImage` method that's implemented into the `SpriteSheet` class is also called by each of the derived classes when attempting to retrieve an image from the loaded sprite sheet. Because each image in the loaded sprite sheet is the same width and height, these values remain constant and it's just the x- and y-position of the image that changes, so the correct image on the sprite sheet is captured. The `PacmanSprites` class has a function called `defineAnimations` that's used to set up *Pac-Man* movement animations in all directions: left, right, up and down and ultimately for when *Pac-Man* dies from being caught by one of the ghosts.



Most of the dots have been cleared, but Pac-Man's lost three lives in the process!

The Pacman class

Let's now look at the main `Pacman` class that's inherited from a class called `Entity`. With your chosen editor, open up a Python script file called **pacman.py**. This class is mainly used to monitor and control the *Pac-Man* game character.

```
class Pacman(Entity):
  def __init__(self,node):
    Entity.__init__(self,node)
    self.name = PACMAN
    self.color = YELLOW
    self.direction = LEFT
    self.setBetweenNodes(LEFT)
    self.alive = True
    self.sprites = PacmanSprites(self)
```

As can be seen from the above code, the constructor of the `Pacman` class is used to set up the starting direction of *Pac-man* (in this case facing to the left) and various other details, as well identifying that *Pac-man* is alive and setting the value for `Pacman`. Also, the constructor of the `Entity` class is called within the `Pacman` class constructor.

The references to names in capital letters (usually referred to as constants) are stored in a Python script file called **constants.py**, which we'll look at later – just in case you were wondering where for example `PACMAN` or `LEFT` came from. If you'd like to change the starting direction of *Pac-man* to face the right direction at the start of the game, there are two lines of code that you'll need to change in the constructor, as shown below.

```
self.direction = RIGHT
self.setBetweenNodes(RIGHT)
```

The above code will work when the game is first started, but *Pac-man* will revert to facing the left direction once he's caught by one of the ghosts. To solve this, we need to make a further adjustment in a method called reset.

In your editor, scroll down to where the reset method is and alter the same two lines of code as above in the reset method. After, save your changes and run the game program as before. When the game program is now executed, the *Pac-Man* character will face right and will still face right even after he's killed. The `Pacman` class has other implemented functionality, including:

- Functionality to die when caught by a ghost
- Updating the position and direction of *Pac-Man*
- Detecting collisions
- *Pac-Man* character control

» GAME ANIMATIONS

All the animations for this game have been achieved by collecting images into an array in a set order and then executing them depending on the input from the player of the game. Back in **LXF277** a coding project used the Python math library to animate character movement and actions. This month's project uses a collection of images to animate *Pac-man* and the ghosts. There are 12 images in total to perform all the animation sequences for *Pac-man*. These involve animations for when *Pac-man* character is moving up, down, left or right and also when *Pac-man* dies.

You'll be able to see this for yourself when looking at the sprite image file **spritesheet_mspacman.png**. There are four different images for each ghost character in the game, totalling 16 ghost images. Each of the four ghost images has its eyes in a slightly different location to give the impression the eyes are moving when the ghost character is moving around the maze. Again, the images are cycled through to create the illusion that the eyes are moving.

The control of the *Pac-Man* character is implemented in a method called `getValidKey`. While in your editor, scroll down to where this method is implemented:

```
def getValidKey(self):
    key_pressed = pygame.key.get_pressed()
    if key_pressed[K_UP]:
        return UP
    if key_pressed[K_DOWN]:
        return DOWN
    if key_pressed[K_LEFT]:
        return LEFT
    if key_pressed[K_RIGHT]:
        return RIGHT
    return STOP
```

The current way of controlling the *Pac-Man* character is by keyboard. This can be changed to another control method such as an XBox video game controller. This was recently covered and implemented in **LXF292**. Following this tutorial, it's possible to change from using the cursor keys to using the more common W, S, A and D key combination.

A word on constants

A common approach and technique in coding is to use a variable to represent constant values. These are usually written in uppercase that are referred to through a programs operation. For this project, all constant values are kept in a Python script file called **constants.py**. With your chosen editor, find and open the Python file **constants.py**. As you look down the script, you'll notice constant variables that represent different colours, specify directional movements, specifying object identifiers as well as other aspects such as screen resolution aspects such as width and height.

The ghosts

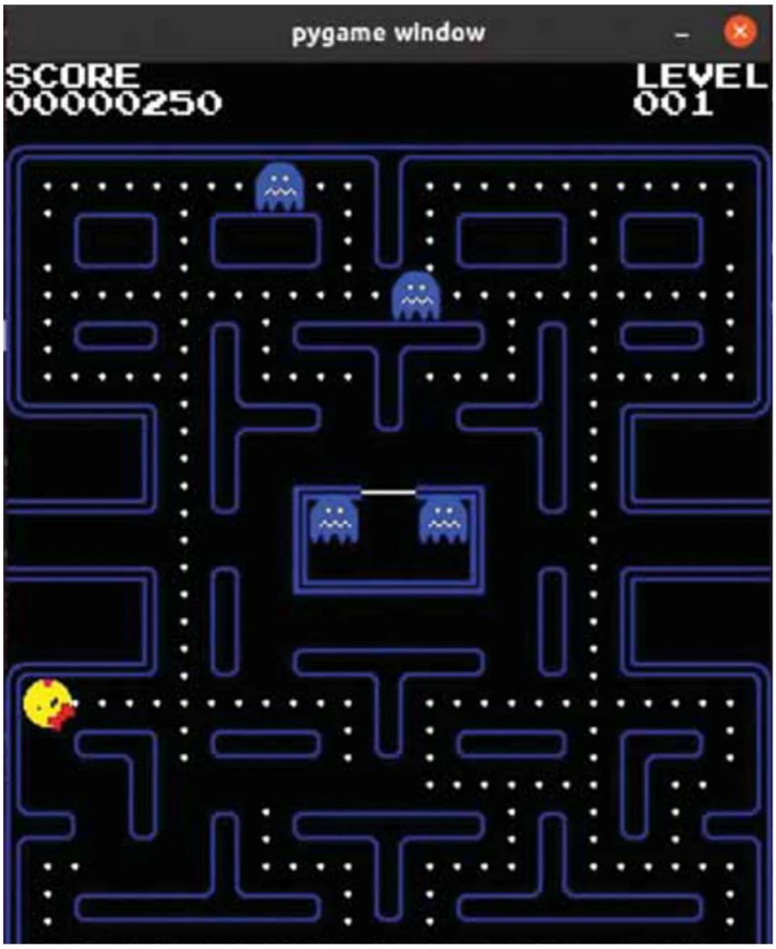
Finally we'll now look more closely at how the ghost characters have been implemented for this game. With your chosen editor, open up the Python script file **ghosts.py**. You'll see from viewing the code in **ghosts.py** that the implementation is similar to that of **pacman.py**. At the top of the script file there's a base class called `Ghost`, which the classes implemented below this are inherited from. There are derived classes that are named after the ghosts themselves: Blinky, Pinky Inky and Clyde. Let's take a look at one of these classes. Scroll down to the class called `Blinky`.

```
class Blinky(Ghost):
    def __init__(self, node, pacman=None, blinky=None):
        Ghost.__init__(self, node, pacman, blinky)
        self.name = BLINKY
        self.color = RED
        self.sprites = GhostSprites(self)
```

As you can see from the above code, there's not much code to this class implementation because much of the properties and functionality have been implemented from the `Ghost` class. Looking at the constructor of Blinky, you'll see that it calls the constructor of the `Ghost` class on initialisation. Code is then written to identify it as the Blinky ghost character and continues to specify the colour of Blinky and the sprite image collection.

Have fun reliving this classic 80s arcade game! **LXF**

The four ghosts – Blinky, Pinky Inky and Clyde – are on the run after Pac-man tucked into a power pill.



» GET THE GHOST OF LXF PAST...Subscribe now at **http://bit.ly/LinuxFormat**

YAD

Credit: https://github.com/v1cont/yad

Add useful dialogs to terminal scripts

Quickly create dialogs for your shell scripts and even add notifications or menus with the help of **Karsten Günther**.

OUR EXPERT

Karsten Günther is a long-time LaTeX, Linux and GIMP user, and has written a number of books on these topics.

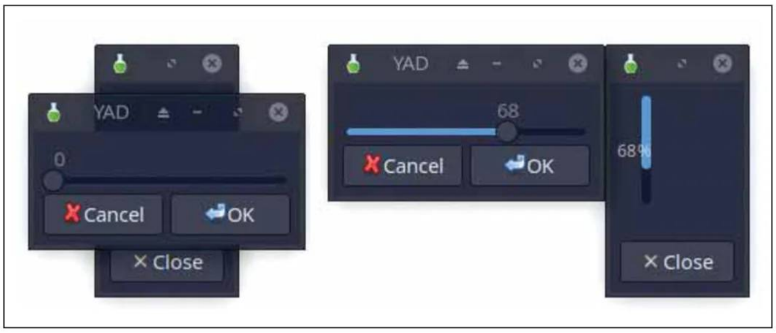
There are some dialog programs for the shell, most of which are based on *Dialog* and *Zenity*, such as *Yad*. This program is optimised for direct use on BASH and can accept input as well as display output. Here, we'll first look at some very simple examples.

The main advantages of *Yad* are the many dialog forms that the program supports – over 20, in fact – and that the developer continuously maintains it. Currently, version 12 is up to date and can be found in the repositories of most distributions.

Yad is controlled by three types of options. First, a "mode" is defined in which *Yad* should work. `yad -h` lists the possible modes. Next, options specific to the selected mode control the exact behaviour of *Yad*. For file selection, typing `yad --file --multiple` enables you to select more than one file at a time.

Which options are available in a mode should be shown by `yad --help-MODUS`, but in many cases the information is incomplete. For example, `yad --help-file` omits the `--multiple` option. These extra options are described in the man page.

The third group of options relates to the window opened by *Yad* and determines its size, position, labelling and so on. While only one of the options for the mode can be selected at a time, the supplementary options can, of course, be used as desired and according to the requirements. The order of the options is flexible to a large extent, but it's good practice to group them systematically in order to keep an overview.



When the command line is launched, the windows are on top of each other. Moving the horizontal slider shows the effect in the vertical one.

Sliding scales

A minimalist example uses the sliders to set and display numerical values. The code is very clear: `yad --scale --print-partial | yad --progress --vertical`. The first part creates a slider (`yad --scale --print-partial`) with which numerical values can be set, without additional options in the range from 0 to 100. `--print-partial` instructs the command to output the values immediately and not wait for the dialog to end. The pipe symbol connects this command with a second one that, because of the option `--progress`, creates an indicator that looks like the slider and is arranged vertically for differentiation.

Because no position information has been passed to the *Yad* commands, the windows initially appear on top of each other. Try moving a window with the mouse to see the effect. The result are shown in the screenshot (*below left*). If the horizontal slider is moved up and down, the progress bar instantly follows.

The example can be improved in many ways. With `--width=WIDTH` and `--height=HEIGHT` the windows can be placed, and the sliders and the progress bar can be configured precisely, too. See the program's man page for detailed information.

User menus with notifications

A highlight of *Yad* is using real desktop menus with the "notification" mode. You'll be aware of many programs that appear in the "notification area" of the panels, for example, when updates are available. A mouse click on the displayed icon starts the update.

Yad displays an icon in the panel through which the program communicates. *Yad* supports actions for a left-, middle- or right-mouse click on the icon. By default, a left- or middle-mouse click will terminate *Yad*, unless other actions are set with options. When you type

```
--command="COMMANDLINE"
```

Yad executes the specified command line on each left mouse click. Meanwhile, `--no-middle` prevents *Yad* from exiting on a middle mouse click. A simple command line outputs `left` on the terminal, if the *Yad* icon is clicked with the left-mouse button, and it displays `menu` by pressing the right-mouse button:

```
 yad --notification --command='echo left'
--menu='MENU'
```

What's on the menu?

The option `--menu='MENU'` creates a menu. Menu entries are defined line by line. Each consists of several parts. A `MENU TEXT` is obligatory, a command line to be executed makes sense, but isn't necessary, and optionally `ICON` (more below on this)can be used. *Yad* uses an exclamation mark by default as a separator between the parts (use `--separator='CHARACTER'` to change this).

The menu definitions must be quoted for Bash to recognise it correctly. If another menu item is added then it's separated by a pipe symbol (to change this use `--item-separator='CHARACTER'`), like so:

```
 yad --notification --menu='MENU1!commandline
1|MENU2!commandline 2'
```

It's should be obvious that such command lines will quickly become confusing. There's a second method to communicate with *Yad* via the FIFO method (first in, first out). *Yad* uses the option `--listen` for this. If it's specified then *Yad* accepts data via the standard input channel.

Note that the syntax is slightly extended. First, a keyword must be sent, which determines what data follows. Currently, six of these keywords are available:

```
 menu:MENUTEXT!COMMANDLINE!MENUICON|...
```

creates the menu items, in the form described above.

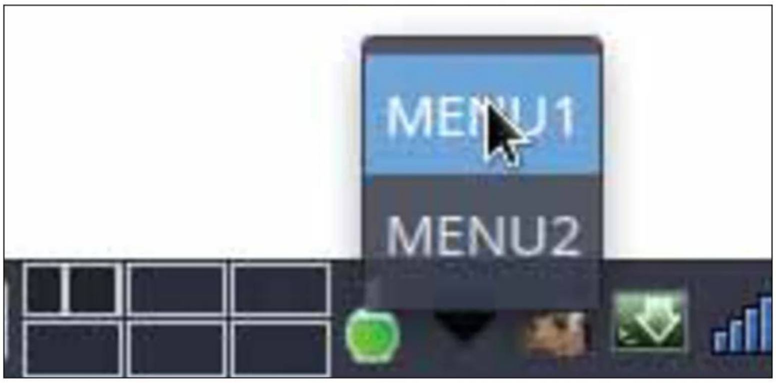
Action plan

The keyword `action:` specifies the command to be executed by a left-mouse click. There are two special commands: `menu` shows the defined menu; while `quit` causes *Yad* to exit.

```
 icon:PATH/TO/THE/ICON
```

sets the *Yad* icon displayed in the panel. `tooltip:"TEXT FOR THE TOOLTIP"` defines a tooltip.

The keyword `visible:HOW` defines how the icon is displayed. Three types are provided: `true` displays it, `false` hides it and `blink` makes it flicker.



Yad creates its menus in this area of the desktop environment with a simple command line.

QUICK TIP

Make your life easy and use the menu template to create your own menus. Only the lines including the ... have to be adapted.

The following code is based on an example published by *Yad* developer Victor Ananjevsky himself. This variant uses an additional data channel:

1) a FIFO is set up: `mkfifo FIFO`

2) the FIFO is established as an additional channel: `exec 3<> FIFO`

3) *Yad* is started in the background connected to the FIFO: `yad --notification --listen <&3 &`

4) Settings are sent to *Yad* using `echo >&3`

Because the menu items are given as arguments to `echo`, the normal rules of the shell apply here. Line breaks can be used as long as they're masked. Indentations at the beginning of lines leads to a shifted display of the menu items.

Options such as `--command=` or `--no-middle` must be used when calling *Yad* (`yad --notification --no-middle ...`).

The box (*below*) shows the whole script (**mymenu.sh**), which creates a user menu by *Yad*. There are some special features to note here. `FIFO=$(mktemp)` creates a random, unique file name under `/tmp/`. If a file with this name already exists, it will be deleted. With this name, `mkfifo` creates a special **Fifo** file ("named pipe") and is used to communicate with *Yad*.

Yad is launched with the option `--command='...'` to prevent the program from terminating on a left-mouse click. The menu itself is defined via `echo`.

The penultimate line of the menu definition has a special feature: `...fin!quit` creates the menu entry `...fin` and terminates *Yad* when clicked.

Finally, note that the definitions of the tooltip text and for the notification icon are comparable to the menu definition. **LXF**

» CREATING USER-DEFINED MENUS WITH YAD

Here's our **mymenu.sh** example test script, install YAD and pop this into a text editor to give it a try!

```
#!/bin/bash
# save as: /usr/local/bin/mymenu.sh
# create FIFO
FIFO="$(mktemp)"
rm $FIFO
mkfifo $FIFO

# establish channel
exec 3<> $FIFO

# start yad
yad --notification --command='xterm'
--listen <&3 &

# define menus
>&3 echo "menu:\
...! start here|\
EOG!eog|\
Terminal!xterm|\
EDIT Menu!gedit /usr/local/bin/
mymenu.sh|\
X-TILE 1!x-tile 1|\
...fin!quit \
"

# define tooltip and icon
>&3 echo "TOOLTIP:My TOOLTIP"
>&3 echo "icon:/usr/share/icons/
Adwaita/16x16/emblems/emblem-
favorite-symbolic.symbolic.png"
```

EBPF

Credit: https://ebpf.io

Part One! Don't miss next issue! Subscribe on page 16

Getting started with the powerful eBPF

Extending things is what **Mihalis Tsoukalos** loves to do, so his heart leapt at the chance to explain the Extended Berkeley Packet Filter.

OUR EXPERT

Mihalis Tsoukalos is a systems engineer and a technical writer. You can reach him at www.mtsoukalos.eu and @mactsouk.

The subject of this tutorial is eBPF and is the first in a series of tutorials about eBPF and its capabilities. eBPF stands for Extended Berkeley Packet Filter, but despite its name it's not restricted to just network packet filtering.

More accurately, eBPF enables users with root privileges to load and execute custom programs inside the kernel of the operating system, which, in simple terms, is at a really low level. After a custom program is loaded, it's verified for safety and it might be rejected if there are problems with it. However, if everything is okay then the program is attached to at least one event (this is a compulsory requirement) and is launched when that event takes place.

As a consequence, instead of changing our software and adding extra code to it, we can collect information about our software by using an existing eBPF tool or by writing our own eBPF tools.

The history of eBPF

Let's first discuss the history of BPF and eBPF, and how they're connected. BPF was introduced in 1992 for improving the performance of packet capture tools. In 2013 Alexei Starovoitov did a major rewrite of BPF that was included in the Linux kernel in 2014 and replaced BPF. With this rewrite, BPF, which is now called eBPF, became more versatile and can be used for a variety of tasks beyond simple network packet capture.

It's difficult to describe precisely what eBPF can do because it has so many capabilities. It's much easier to describe how we can use eBPF. It can be used in three main areas: networking, security and observability. This series of tutorials is going to focus on the observability capabilities (tracing) of eBPF.

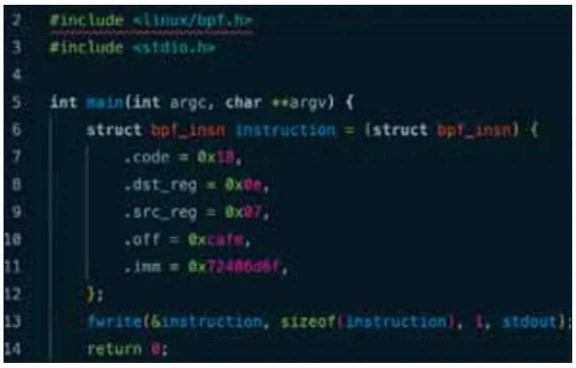
The main advantage of eBPF compared to traditional performance tools is that it's efficient, production safe and part of the Linux kernel. In practice, this means that we can use eBPF without the need to add or load any other components to the Linux kernel.

Most Linux applications are executed in user space, which is a layer without too many privileges. Although using user space is safer and more secure, it has restrictions and requires using system calls to ask the kernel for access to privileged resources. Even the simplest commands use a large amount of system calls when executed.

In practice, this means that if we're able to observe the system calls of our applications, we can learn more information about the way they behave and operate. When things operate as expected and the performance of our applications is good, we usually don't care much about performance and the executed system calls. However, when things go wrong we're keen to understand more about the operation of our applications, and solve the problem.

QUICK TIP

Get the code for this tutorial from the Linux Format archive: www.linuxformat.com/archives?issue=294.



```
#include <linux/bpf.h>
#include <stdio.h>

int main(int argc, char **argv) {
    struct bpf_insn instruction = (struct bpf_insn) {
        .code = 0x18,
        .dst_reg = 0x0e,
        .src_reg = 0x07,
        .off = 0xcafe,
        .imm = 0x72486d6f,
    };
    fwrite(&instruction, sizeof(instruction), 1, stdout);
    return 0;
```

This screenshot shows the C code of simple.c, which just defines an eBPF C structure.

Behind the scenes

Putting special code in the Linux kernel or developing a module to understand the operation of our applications is a difficult task that might require a long period of time. This is where observability and eBPF come into play. eBPF, its language and its tools enables us to dynamically see what happens behind the scenes without the need to change the Linux operating system.

All you need to communicate with eBPF is a programming language that supports libbpf (see

https://github.com/libbpf/libbpf). Apart from C, the Go programming language offers support for libbpf (details at **https://github.com/aquasecurity/libbpfgo**). However, discussing Go support for eBPF is beyond the scope of this tutorial.

The next section shows how to install eBPF and the BCC project on an Arch Linux machine and how to make sure that the eBPF installation is working properly.

Installing eBPF and BCC

A useful resource when using eBPF is the BCC project, which comes with several handy tools and examples that use extended BPF. Among other things, BCC enables us to write BPF programs using Python and Lua. However, kernel instrumentation is still implemented in C or C++. So, let's make sure that we have eBPF enabled, BCC installed and that we can use them on our Linux machine.

In order to install all of these on an Arch Linux system we should execute `pacman -S bpf bcc bcc-tools python-bcc` with root privileges. For other distros see: **https://github.com/iovisor/bcc/blob/master/INSTALL.md**. The last package is about working with Python and BCC and isn't essential, but it is going to be used later on in this tutorial.

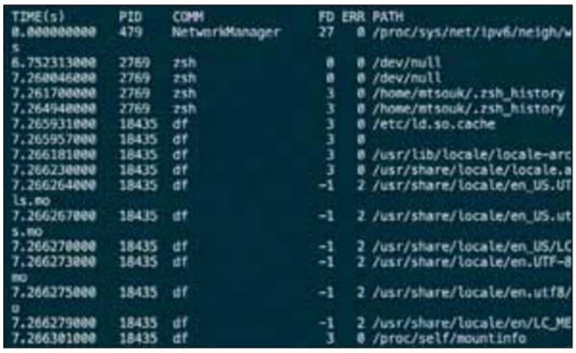
To make sure that everything is working correctly, execute the `bpftrace` command as `bpftrace --version`, to learn more about its version. If the output looks similar to `bpftrace v0.14.1`, then we're ready to go.

Before continuing, bear in mind that most of the tools and code in this tutorial require administrative privileges. This makes perfect sense because regular users shouldn't be able to interact with the Linux kernel directly. The `bpftrace --version` command is an exception to that rule.

Using C

This section is going to present a simple C source code file that uses an eBPF structure to make sure that everything is okay with our Linux machine and that all required libraries and C header files are present. The screenshot (*left*) shows the entire code of **simple.c**. Even for such a small C program, this is a pretty low-level code that's difficult to read and maintain, especially if you're not familiar with C or C++.

The program makes use of an eBPF structure named `bpf_insn` that can be found in **linux/bpf.h**. Note that all C programs which work with eBPF need to define similar structures.



```
TIME(s)       PID    COMM              FD ERR PATH
0.000000000   479    NetworkManager    27   0 /proc/sys/net/ipv6/neigh/w
s
6.752313000   2769   zsh                0   0 /dev/null
7.260046000   2769   zsh                0   0 /dev/null
7.261700000   2769   zsh                3   0 /home/mtsouk/.zsh_history
7.264940000   2769   zsh                3   0 /home/mtsouk/.zsh_history
7.265931000   18435  df                 3   0 /etc/ld.so.cache
7.265957000   18435  df                 3   0
7.266181000   18435  df                 3   0 /usr/lib/locale/locale-arc
7.266230000   18435  df                 3   0 /usr/share/locale/locale.a
7.266264000   18435  df                -1   2 /usr/share/locale/en_US.UT
ls.mo
7.266267000   18435  df                -1   2 /usr/share/locale/en_US.ut
s.mo
7.266270000   18435  df                -1   2 /usr/share/locale/en_US/LC
7.266273000   18435  df                -1   2 /usr/share/locale/en.UTF-8
mo
7.266275000   18435  df                -1   2 /usr/share/locale/en.utf8/
o
7.266279000   18435  df                -1   2 /usr/share/locale/en/LC_ME
7.266301000   18435  df                 3   0 /proc/self/mountinfo
```

This screenshot shows the type of output the `opensnoop -T` command generates. The first column is the timestamp, starting from 0 in the first entry.

The computing overhead of including BPF-related code in your C source files is big for both small C programs and larger ones. Therefore, writing C code might be a problem in most cases, mainly because of the issues that come with C and the extra code that we need to write for supporting eBPF.

Compiling and running **simple.c** generates the next kind of output when piped through the *xxd* utility:

```
$ gcc simple.c -o simple
$ ./simple | xxd
00000000: 187e feca 6f6d 4072  .~..om@r
```

You can learn a lot more about working with C and eBPF at **https://bolinfest.github.io/opensnoop-native**. The next section shows how to use eBPF with Python, which because of the integration offered by BCC enables you to write applications using less code than when working with C or C++.

Using Python

Similarly to the previous section, this section presents a Python script that uses eBPF. Before continuing, please install the bcc Python package by running `pip3 install bcc` if it's not already installed. The following code prints a message on the screen every time a new process is generated. The code of **simple.py** is as follows:

```
from bcc import BPF
BPF(text="""
  int kprobe__sys_clone(void *ctx) {
    bpf_trace_printk("New process!!\\n");
    return 0;
  }""").trace_print()
```

Note that if you try to run **simple.py** without root privileges, you're going to receive an error message along the lines of `cannot import name 'BPF' from 'bcc'`. Assuming that you've already installed the bcc Python 3 package, the cause of the error message is simply

QUICK TIP

eBPF isn't the only tool and certainly not the first tool in its category. The first such tool was DTrace, which is a comprehensive dynamic tracing framework. It first appeared in Sun Solaris. Nowadays, you can find DTrace available in macOS and other BSD variants.

» THE BCC PROJECT

The BCC project is built on top of eBPF and offers tools as well as interfaces for higher-level programming languages such as C, C++, Lua and Python. BCC saves you from having to write lower-level code using BPF instructions, and is compiled using LLVM or LLVM Intermediate Representation.

Nowadays, most eBPF tools are programmed in BCC or bpftrace. BCC is better at building complex tools with many command line arguments and options that are fully customisable whereas bpftrace is better at one liners and shorter tools. Developing in BCC takes 10 times longer and requires 10 times more code than using bpftrace!

The most important and handy BCC tools are going to be presented in more detail in forthcoming *Linux Format* eBPF tutorials, in which we'll explore specific issues. What you should remember here is that if there is an existing tool that does your job, you should try to use that. Otherwise, you might need to use the bpftrace language and the *bpftrace* utility, or write our own code in one of the supported programming languages.

The home of the BCC project is at **https://github.com/iovisor/bcc**. You can find analytical installation instructions at **https://github.com/iovisor/bcc/blob/master/INSTALL.md**.

»

QUICK TIP

Although eBPF was originally developed for Linux, Microsoft is developing a version of eBPF that runs under Microsoft Windows. The rate of eBPF development means it might become the default Networking, Security, and Observability framework.

because you're not executing **simply.py** with root privileges. This should be easy to resolve.

Let us discuss some important details. The BPF() object does the job of compiling and running the BPF code. The value of `text` defines the BPF program that we want to run – this is C code.

`trace_print()` is a BCC function that reads the generated output, which in our case is generated by `bpf_trace_printk()`, and prints it on screen. The `kprobe__sys_clone()` function captures newly generated processes. The `kprobe` preamble identifies that we want to perform kernel dynamic tracing via `kprobes`, and we're going to learn more about `kprobes` in next month's *Linux Format*.

The remainder of `kprobe__sys_clone()`, which is `sys_clone()`, identifies the name of the `kprobe`, which is just `sys_clone`. This means that every time the `sys_clone()` kernel function (**https://linux.die.net/man/2/clone**) is called, the `kprobe__sys_clone()` function of our program is executed.

Running **simple.py** generates the following output:

```
b' zsh-2769          [002] d..31 193865.870274: bpf_trace_printk: New process!!'
b' skypeforlinux-1780   [007] d..31 193877.670598: bpf_trace_printk: New process!!'
```

The first output is from the zsh shell whereas the second output is from *Skype for Linux*. You should now press Ctrl+C for the **simple.py** Python script to end. If you don't then you're going to see lots of output on your screen, especially if you're working on a busy Linux machine.

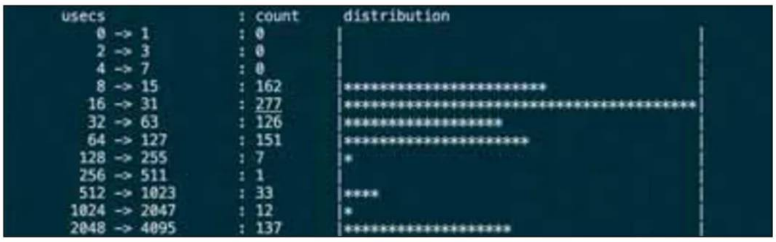
The rest of this tutorial is going to present some of the handy tools that come with the BCC project as well as some one liners that use the bpftrace language. But before that, a word of warning…

Most BCC tools are located in the **/usr/share/bcc/tools** directory, at least on an Arch Linux machine, which isn't included in the PATH environment variable by default. This means that you should execute all these commands using their full path. Alternatively, you should run `cd /usr/share/bcc/tools` first. Additionally, most of the tools are either Python scripts or bash shell scripts, which means that you can look into them or even modify their code.

The opensnoop utility

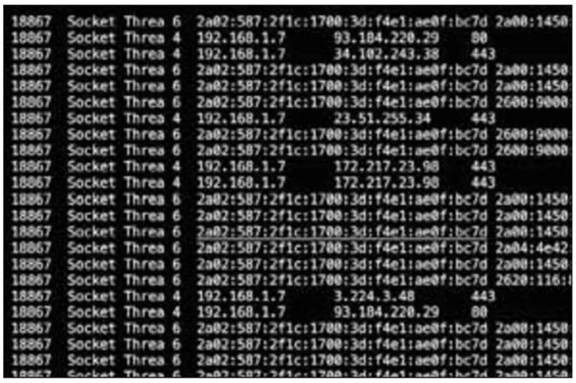
We're going to begin by presenting a basic tool called *opensnoop* that enables you to see the files that are opened on your Linux system. Essentially, *opensnoop* looks for calls to the `open()` system call. The generated *opensnoop* output looks like the following:

```
0.006969000 1737  ThreadPoolForeg   28  0 /home/mtsouk/.config/skypeforlinux/Cache/f74a1a87d4226ac0_0
0.007046000 1737  ThreadPoolForeg   -1  2 /run/systemd/machines/static-asm.secure.skypeassets.com
1.036388000 1737  ThreadPoolForeg   31  0 /etc/hosts
```



```
     usecs               : count     distribution
         0 -> 1          : 0        |                                        |
         2 -> 3          : 0        |                                        |
         4 -> 7          : 0        |                                        |
         8 -> 15         : 162      |***********************                 |
        16 -> 31         : 277      |****************************************|
        32 -> 63         : 126      |******************                      |
        64 -> 127        : 151      |*********************                   |
       128 -> 255        : 7        |*                                       |
       256 -> 511        : 1        |                                        |
       512 -> 1023       : 33       |****                                    |
      1024 -> 2047       : 12       |*                                       |
      2048 -> 4095       : 137      |*******************                     |
```

This screenshot presents the output you should expect to obtain from the biolatency tool that traces disk I/O latency.



```
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
18867  Socket Threa 4  192.168.1.7         93.184.220.29      80
18867  Socket Threa 4  192.168.1.7         34.102.243.38      443
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2600:9000
18867  Socket Threa 4  192.168.1.7         23.51.255.34       443
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2600:9000
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2600:9000
18867  Socket Threa 4  192.168.1.7         172.217.23.98      443
18867  Socket Threa 4  192.168.1.7         172.217.23.98      443
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a04:4e42
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2620:116:
18867  Socket Threa 4  192.168.1.7         3.224.3.48         443
18867  Socket Threa 4  192.168.1.7         93.184.220.29      80
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
18867  Socket Threa 6  2a02:587:2f1c:1700:3d:f4e1:ae0f:bc7d 2a00:1450
```

This screenshot shows the output generated by the tcpconnect utility, which traces the connect(2) system call in order to identify active TCP connections.

The output of the third column shows the command that used the `open()` call, while the output of the fifth column shows the return value of the call. In UNIX, 0 means no error whereas other values might indicate error conditions, bytes read or written, and so on.

Usually *opensnoop* is used for locating the paths of configuration files or for understanding why a program fails to be properly executed for reasons related to configuration files, external libraries or other system files. Even if you're not interested in troubleshooting a process, you can still use *opensnoop* to understand what's happening behind the scenes on your Linux machine. The screenshot (*page 95*) shows a more detailed output of *opensnoop*.

The *biolatency* tool displays the disk I/O latency and calculates a histogram of the results. The screenshot (*below left*) shows an example of the output you should expect to obtain from the *biolatency* tool.

The *tcpconnect* tool traces the `connect()` system call in order to report all active TCP connections. The screenshot (*above*) shows the type of output generated by *tcpconnect*, which is much better than the output of *netstat*. What's interesting is that the utility displays the IP address of both the source and the destination address as well as the port number used.

The *execsnoop* utility captures the execution of new processes in an entire Linux system by tracing `execve()` calls. It's handy for finding out what gets executed on your Linux machine. The screenshot (*top of page 97*) shows the type of output you should expect to obtain from *execsnoop*.

The bpftrace language

This final section of this tutorial is going to present some simple uses of the *bpftrace* utility, along with the language to familiarise you with how it works. As mentioned elsewhere in this tutorial, the bpftrace language is heavily inspired by AWK and C. The logic of the bpftrace language is based on probes. Put simply, a probe is what triggers the operation of eBPF and its tools. Strictly speaking, a probe is a software or hardware instrumentation point that can generate events which can be caught and therefore handled by eBPF code.

The first thing to be aware of is that you can obtain the full list of supported probes by running `bpftrace -l` as root. The list is big and contains more than 60,000

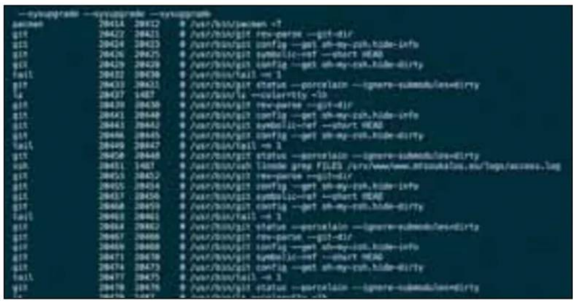
» MORE ABOUT EBPF

It's important to be able to observe the operation of an existing application without changing its implementation, and that is the main reason behind the creation of eBPF and all similar tools. If you want to learn more about eBPF, you should definitely visit the website of computing expert Brendan Gregg at **www.brendangregg.com**.

eBPF has its own high level language, which is named bpftrace and is available in all modern Linux kernels. Bpftrace, which is created by Alastair Robertson, uses both LLVM for compiling scripts to BPF-bytecode and BCC for interacting with the Linux BPF system. You can find more information about bpftrace at **https://github.com/iovisor/bpftrace**. You can use the bpftrace language with the help of the bpftrace binary (please type "man bpftrace" to learn more about it).

If the available tools don't cover the task that you want to do, then you're going to need to write some bpftrace code that meets your requirements. The good thing is that the bpftrace language looks a lot like C and AWK, and that most of the times a small amount of bpftrace code is going to perform the desired task. So, with the use of bpftrace you can create your own tools just as the BCC project did.

Finally, the reference guide of the bpftrace language can be found at **https://github.com/iovisor/bpftrace/blob/master/docs/reference_guide.md**.



This screenshot shows the kind of output generated by the execsnoop utility, which traces the execve(2) system call.

entries. If you're looking for something in the `bpftrace -l` output, feel free to use *grep* as a filter.

Now it's time to present some simple examples of the bpftrace language as a demonstration of what the language can do. Note that running bpftrace with the `-e` flag enables you to write bpftrace code inline in a similar manner to Perl or Python, which is pretty handy for one-liners. However, if you want to keep your BPF code in a separate file, you can use `bpftrace <filename>` to execute it. By conversion, bpftrace source code files have the .bt file extension.

The first one-liner is the following:

```
$ bpftrace -e 'software:page-faults:1 { @[comm,pid] = count(); }'
```

This code counts the page faults by process. The following is a small extract from the output from the previous one-liner:

```
@[zsh, 20756]: 57
@[zsh, 20753]: 66
@[git, 20757]: 213
@[python3, 20721]: 10203
```

Each line shows the name of the executable, its process ID and the number of page faults. Having processes with a very large number of page faults is bad for the performance of your Linux system.

Among other things, bpftrace comes with support for calculating counts, summaries, averages, minimum, maximum, statistics and generating histograms. The following bpftrace code creates a histogram based on the `read()` call request sizes:

```
$ bpftrace -e 'tracepoint:syscalls:sys_enter_read { @ = hist(args->count); }'
```

Part of its output should look as follows:

```
[1]          17 |@@                |
[2, 4)        28 |@@@              |
[4, 8)         4 |                 |
[8, 16)      137 |@@@@@@@@@@@@@@@@@|
```

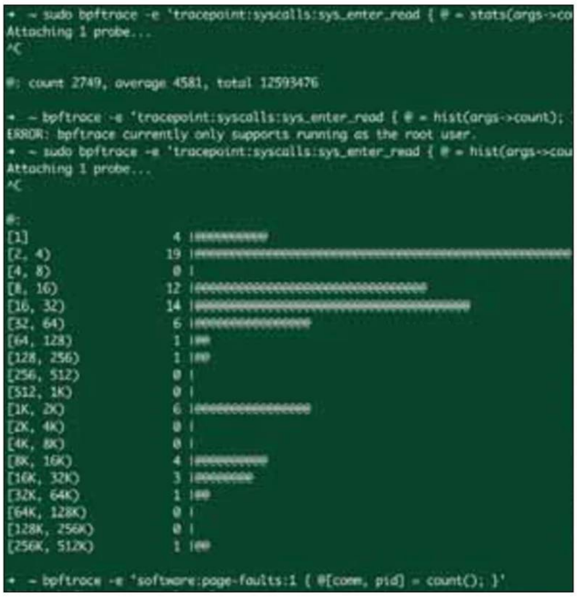
The last one-liner displays summary statistics regarding `read()` system calls:

```
$ sudo bpftrace -e 'tracepoint:syscalls:sys_enter_read { @ = stats(args->count); }'
@: count 2012, average 870, total 1751513
```

The previous output means that we processed 2,012 calls, the average value was 870 and the summary of all values was 1,751,513.

The screenshot (*below*) shows some of the output from the previous bpftrace examples. You should run all of them on your own machine and study the output.

In this tutorial you have learned how to install eBPF and the tools for the BCC project, how to use eBPF with C and Python, learned about some basic BCC tools and the *bpftrace* utility and language. Next month's tutorial is on working with eBPF. Until then, remember that eBPF presents many observation possibilities. However, because eBPF is a complex tool, you'll need to learn it before being able to use it on production systems. **LXF**



```
➜  ~ sudo bpftrace -e 'tracepoint:syscalls:sys_enter_read { @ = stats(args->co
Attaching 1 probe...
^C

@: count 2749, average 4581, total 12593476

➜  ~ bpftrace -e 'tracepoint:syscalls:sys_enter_read { @ = hist(args->count);
ERROR: bpftrace currently only supports running as the root user.
➜  ~ sudo bpftrace -e 'tracepoint:syscalls:sys_enter_read { @ = hist(args->cou
Attaching 1 probe...
^C

@:
[1]                    4 |@@@@@@@@@@
[2, 4)                19 |@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
[4, 8)                 0 |
[8, 16)               12 |@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
[16, 32)              14 |@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
[32, 64)               6 |@@@@@@@@@@@@@@@@
[64, 128)              1 |@@
[128, 256)             1 |@@
[256, 512)             0 |
[512, 1K)              0 |
[1K, 2K)               6 |@@@@@@@@@@@@@@@@
[2K, 4K)               0 |
[4K, 8K)               0 |
[8K, 16K)              4 |@@@@@@@@@@
[16K, 32K)             3 |@@@@@@@@
[32K, 64K)             1 |@@
[64K, 128K)            0 |
[128K, 256K)           0 |
[256K, 512K)           1 |@@

➜  ~ bpftrace -e 'software:page-faults:1 { @[comm, pid] = count(); }'
```

This screenshot shows a part of the output from some handy bpftrace one liners. It's impressive how much you can do with some simple bpftrace code.

QUICK TIP

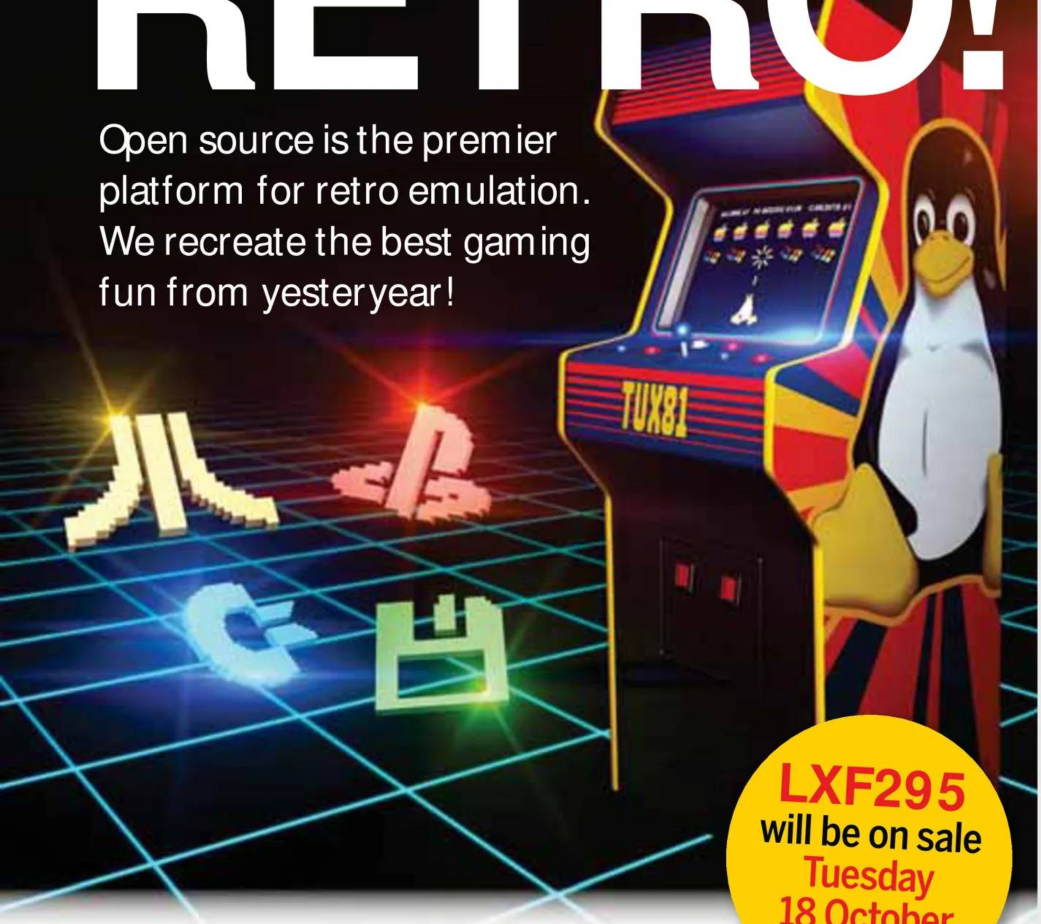
You can find the BCC reference, which contains low-level information, at https://bit.ly/lxf294bcc. The /usr/share/bcc directory offers lots of information about the BCC project. Note that the relevant code is located on GitHub in a Linux Foundation project that's called iovisor, hence the presence of the iovisor word in the two GitHub paths..

» **KEEP A TRACE ON US EACH ISSUE** Subscribe now at **http://bit.ly/LinuxFormat**

NEXT MONTH

GOING FULL RETRO!

Open source is the premier platform for retro emulation. We recreate the best gaming fun from yesteryear!



LXF295 will be on sale Tuesday 18 October 2022

Lightweight distros

Faster to respond, quicker to load, lower memory use… we put the latest lightweight Linux distributions to the test.

Full-disk encryption

Protect your entire disk from snooping and data loss with our easy-to-follow guide to encrypting entire drives!

Run a mainframe!

You don't need a spare basement with a separate industrial-level power supply, just some COBOL and open source.

Windows Subsystem for Linux

We explore how the new WSL 2.0 works and what it can do for Linux and running open source software on Windows systems.

Contents of future issues subject to change – because it doesn't get more retro than print publishing.

LINUX FORMAT
The #1 open source mag

Future Publishing Limited,
Quay House, The Ambury, Bath, BA1 1UA
Email linuxformat@futurenet.com

EDITORIAL
Editor Neil Mohr
Grub editor Jonni Bidwell
Art editor Efrain Hernandez-Mendoza
Operations editor Cliff Hope
Group editor-in-chief Graham Barlow
Group art director Jo Gulliver
Editorial contributors
Mike Bedford, Neil Bothwick, Matthew Castle, Alan Dexter, Karsten Günther, Matthew Hanson, Matthew Holder, Alastair Jennings, Jon Masters, Nick Peers, Les Pounder, Michael Reed, David Rutland, Mayank Sharma, Shashank Sharma, Andrew Smith, Mihalis Tsoukalos
Cover illustration magictorch.com
Ubuntu is a trademark of Canonical Limited. We are not endorsed by or affiliated with Canonical Limited or the Ubuntu project. Linux Mint is copyrighted 2022 and trademarked through the Linux Mark Institute. Fedora and the Infinity design logo are trademarks of Red Hat, Inc. Raspberry Pi is a trademark of the Raspberry Pi Foundation. Tux credit: Larry Ewing (lewing@isc.tamu.edu) and The GIMP. Microsoft Windows is a trademark of the Microsoft group of companies.

ADVERTISING
Commercial sales director Clare Dove
clare.dove@futurenet.com
Senior advertising manager Lara Jaggon
lara.jaggon@futurenet.com
Head of commercial – Technology Dave Randall
dave.randall@futurenet.com
Account director Andrew Tilbury
andrew.tilbury@futurenet.com

INTERNATIONAL LICENSING
Head of Print Licensing Rachel Shaw
Linux Format is available for licensing and syndication. To find our more contact us at *licensing@futurenet.com* or view our content at www.futurecontenthub.com.

NEW SUBSCRIPTIONS & BACK ISSUES
Web www.magazinesdirect.com
UK 0330 333 1113 **World** +44 (0) 330 333 1113

EXISTING SUBSCRIPTIONS
Web www.mymagazine.co.uk
UK 0330 333 4333 **World** +44 (0) 330 333 4333
Subscription delays: Disruption remains within UK and International delivery networks. Please allow up to seven days before contacting us about a late delivery to **help@magazinesdirect.com**

CIRCULATION
Head of newstrade Tim Mathers

PRODUCTION AND DISTRIBUTION
Head of production UK & US Mark Constance
Production project manager Clare Scott
Senior ad production manager Jo Crosby
Digital editions controller Jason Hudson

THE MANAGEMENT
MD tech Paul Newman
Content director Joe Cox
Head of art and design Rodney Dive
Design director Brett Lewis
Commercial finance director Dan Jotcham
Printed by Wyndeham Peterborough, Storey's Bar Road, Peterborough, Cambridgeshire, PE1 5YS
Distributed by Marketforce, 5 Churchill Place, Canary Wharf, London, E14 5HU www.marketforce.co.uk
Tel: 0203 787 9001

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries. GNU/Linux is abbreviated to Linux throughout for brevity. Where applicable code printed in this magazine is licensed under the GNU GPL v2 or later. See www.gnu.org/copyleft/gpl.html. All copyrights and trademarks are recognised and respected.
Disclaimer All contents © 2022 Future Publishing Limited or published under licence. All rights reserved. No part of this magazine may be used, stored, transmitted or reproduced in any way without the prior written permission of the publisher. Future Publishing Limited (company number 2008885) is registered in England and Wales. Registered office: Quay House, The Ambury, Bath BA1 1UA. All information contained in this publication is for information only and is, as far as we are aware, correct at the time of going to press. Future cannot accept any responsibility for errors or inaccuracies in such information. You are advised to contact manufacturers and retailers directly with regard to the price of products/services referred to in this publication. Apps and websites mentioned in this publication are not under our control. We are not responsible for their contents or any other changes or updates to them. This magazine is fully independent and not affiliated in any way with the companies mentioned herein.

If you submit material to us, you warrant that you own the material and/or have the necessary rights/permissions to supply the material and you automatically grant Future and its licensees a licence to publish your submission in whole or in part in any/all issues and/or editions of publications, in any format published worldwide and on associated websites, social media channels and associated products. Any material you submit is sent at your own risk and, although every care is taken, neither Future nor its employees, agents, subcontractors or licensees shall be liable for loss or damage. We assume all unsolicited material is for publication unless otherwise stated, and reserve the right to edit, amend, adapt all submissions. All contents in this magazine are used at your own risk. We accept no liability for any loss of data or damage to your systems, peripherals or software through the use of any guide. Notes: everyone's on holiday!!!

We are committed to only using magazine paper derived from responsibly managed, certified forestry and chlorine-free manufacture. The paper in this magazine was sourced and produced from sustainable managed forests, conforming to strict environmental and socioeconomic standards.

Widely Recycled

Future is an award-winning international media group and leading digital business. We reach more than 57 million international consumers a month and create world-class content and advertising solutions for passionate consumers online, on tablet & smartphone and in print.

FUTURE
Connectors. Creators. Experience Makers.

Future plc is a public company quoted on the London Stock Exchange (symbol: FUTR)
www.futureplc.com

Chief executive **Zillah Byng-Thorne**
Non-executive chairman **Richard Huntingford**
Chief financial officer **Penny Ladkin-Brand**

Tel +44 (0)1225 442244

ipso. For press freedom with responsibility

The Electronic Frontier Foundation is the leading nonprofit organization defending civil liberties in the digital world. Founded in 1990, EFF champions user privacy, free expression, and innovation through impact litigation, policy analysis, grassroots activism, and technology development. We work to ensure that rights and freedoms are enhanced and protected as our use of technology grows.

EFF.ORG

ELECTRONIC FRONTIER FOUNDATION

Protecting Rights and Promoting Freedom on the Electronic Frontier

WIN £50
OF NATIONAL BOOK TOKENS

We're giving away £10,000 to spend on books!
Scan the QR code or visit lovepaper.org to enter

PAPER POWER

62% of the energy used to produce paper and paper-based packaging in Europe comes from renewable sources.

Source: Confederation of European Paper Industries (CEPI), 2020.
CEPI represents 92% of European pulp and paper production

Discover the story of paper
www.lovepaper.org
Scan for paper facts, activities, blogs and much more!

LOVE PAPER®
www.lovepaper.org

With thanks to
NATIONAL BOOK tokens

9021
9000
3Jw8GGqdiRsHphQFjEgfZw==